SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY - INFORMATICS

TECHNICAL UNIVERSITY OF MUNICH

Bachelor's Thesis in Information Systems

An Introduction and Overview of Privacy-Enhancing Technologies for Data Processing and Analysis

Jared Fantaye

SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY - INFORMATICS

TECHNICAL UNIVERSITY OF MUNICH

Bachelor's Thesis in Information Systems

An Introduction and Overview of Privacy-Enhancing Technologies for Data Processing and Analysis

Einführung und Überblick über Privacy-Enhancing Technologies für die Verarbeitung und Analyse von Daten

Author:	Jared Fantaye
Supervisor:	Prof. Dr. Florian Matthes
Advisor:	Alexandra Klymenko, M.Sc.
Submission Date:	August 16th, 2023

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Munich, August 16th, 2023 Jared Fantaye

Abstract

Nowadays, data represents a valuable asset collected and analyzed by a wide range of companies in order to gain valuable business insights and create significant business value. While this may benefit businesses and consumers, it could also infringe on individuals' privacy rights. To counteract this kind of privacy invasion, Privacy-Enhancing Technologies (PETs) have been developed to allow organizations to work with sensitive information while still adhering to various data protection laws and regulations. However, several challenges still hinder the widespread adoption of PETs, such as their complexity, a lack of awareness, and the fact that current privacy laws do not mandate the use of these PETs. The combination of these factors has led to a lack of understanding and, therefore, a lack of adoption of Privacy-Enhancing Technologies by the industry.

To address this problem, this thesis aims to provide a comprehensive introduction and overview of the most predominant PETs for data processing and analysis. Its objective is to enable individuals from diverse backgrounds with no prior knowledge in this field to understand better the characteristics, benefits, challenges, and potential use cases of these PETs. To accomplish this, we first conducted a quantitative analysis to identify the five most prevalent PETs in current literature: Federated Learning, Differential Privacy, Homomorphic Encryption, Secure Multi-Party Computation, and Zero-Knowledge Proofs. Intending to obtain a fundamental understanding of each technology, we then conducted a Systematic Literature Review to find the most relevant publications from a large body of work. Based on the synthesized information and the help of several learning frameworks, such as Bloom's Revised Taxonomy and Gagné's Nine Events of Instruction, we could finally develop engaging and easy-to-understand learning content tailored to a non-technical readership. The resulting educational materials can provide a wide range of audiences with a basic understanding of the most prevalent PETs in the current literature.

Kurzfassung

Heutzutage stellen Daten ein wertvolles Gut dar, das von einer Vielzahl von Unternehmen gesammelt und analysiert wird, um wertvolle Erkenntnisse zu gewinnen und einen erheblichen geschäftlichen Nutzen zu erzielen. Dies kann zwar für Unternehmen und Verbraucher gleichermaßen von Vorteil sein, könnte jedoch auch das Recht des Einzelnen auf Privatsphäre verletzen. Um diesem Eingriff in die Privatsphäre entgegenzuwirken, wurden Privacy-Enhancing Technologies (PETs) entwickelt, die es Unternehmen ermöglichen, mit sensiblen Informationen zu arbeiten und gleichzeitig die verschiedenen Datenschutzgesetze und -vorschriften einzuhalten. Jedoch hindern mehrere Herausforderungen noch immer die weit verbreitete Annahme von PETs, wie etwa ihre Komplexität, ein geringer Bekanntheitsgrad und die Tatsache, dass die aktuellen Datenschutzgesetze die Verwendung dieser PETs nicht vorschreiben. Die Kombination dieser Faktoren hat dazu geführt, dass das Verständnis und damit auch die Akzeptanz von PETs in der Industrie derzeit noch unzureichend sind.

Um dieses Problem zu beheben, zielt diese Arbeit darauf ab, eine umfassende Einführung und Übersicht über die am weitesten verbreiteten PETs für die Datenverarbeitung und -analyse zu bieten. Dies soll Personen mit unterschiedlichem Hintergrund und ohne Vorkenntnisse in diesem Bereich ein besseres Verständnis der Eigenschaften, Vorteile, Herausforderungen und Anwendungsmöglichkeiten dieser PETs vermitteln. Um dies zu erreichen, haben wir zunächst eine quantitative Analyse durchgeführt, um die fünf in der aktuellen Literatur am weitesten verbreiteten PETs zu identifizieren: Federated Learning, Differential Privacy, Homomorphic Encryption, Secure Multi-Party Computation und Zero-Knowledge Proofs. Um ein grundlegendes Verständnis jeder Technologie zu erlangen, wurde eine Systematische Literaturrecherche durchgeführt, um die relevantesten Publikationen aus einer großen Anzahl von Arbeiten zu identifizieren. Auf der Grundlage der synthetisierten Informationen und mit Hilfe verschiedener Lernmodelle, wie Bloom's Revised Taxonomy und Gagné's Nine Events of Instruction, konnten wir schließlich ansprechende und leicht verständliche Lerninhalte entwickeln, die auf eine nicht-technische Leserschaft abgestimmt sind. Das daraus resultierende Lehrmaterial kann dazu verwendet werden, einem breiten Publikum ein grundlegendes Verständnis für die in der aktuellen Literatur am häufigsten vorkommenden PETs zu vermitteln.

Contents

1 Introduction

Nowadays, organizations are collecting and analyzing large amounts of data in order to gain valuable insights and create significant business value. As a result, many companies across various industries actively leverage data analytics to improve their services or develop new innovative business models.

For instance, an innovative insurance company, Root Insurance, uses data analytics to offer its customers tailored insurance rates. The company can provide this service by analyzing the respective driving habits of its customers with the help of various sensors in their smartphones. As a result, they can identify patterns of cautious or reckless driving, which in turn helps them provide risk-adjusted offers to their customers. This benefits both the insurance company and its customers, as customers receive fair insurance rates while the insurance company is able to improve its risk management. [1]

Another company that leverages data to improve its services is Netflix, the largest streaming service in the world. In order to maximize its profits, Netflix aims to acquire and retain as many customers as possible. Their recommendation system plays a significant part in achieving this goal, as it uses personalized data to predict what shows or movies a customer might enjoy watching [2]. By providing this personalized service, Netflix has significantly expanded its user base over the past decade. In a paper published in December 2015, Netflix estimated the savings of "the combined effect of personalization and recommendations" [2] at more than one billion dollars a year.

These examples have demonstrated that data represents a valuable asset that can be utilized for a variety of use cases. As a result, data processing has become ubiquitous and is used by a large number of organizations. Although this has numerous benefits for businesses and consumers alike, it also has its pitfalls. Utilizing individuals' personal information without explicit consent violates their privacy and infringes their right to informational self-determination.

To counteract this kind of privacy invasion, Privacy-Enhancing Technologies (PETs) have been developed to allow organizations to work with sensitive information without compromising its privacy. Hence, with the help of these technologies, they can extract valuable insights while still adhering to various data protection laws and regulations. However, existing regulations do not mandate the use of these PETs, leaving it up to individual companies to decide whether or not to use them. As a result, most organizations do not have sufficient incentives to adopt these complex technologies. This, combined with the large volume, complexity, and heterogeneity of information available, leads to a current lack of understanding and adoption of PETs by the industry. Our work aims to address this problem by providing a comprehensive introduction and overview of the most predominant Privacy-Enhancing Technologies tailored to a non-technical audience. This should enable a wide range of stakeholders to gain a fundamental understanding of their characteristics, benefits, challenges, and potential use cases.

However, we would like to emphasize that the scope of this work does not allow us to provide an overview of all Privacy-Enhancing Technologies. For this reason, we will only focus on the five most predominant PETs in the current literature. This should give a non-technical readership an excellent introduction to the PET landscape without being too overwhelming in complexity.

In order to ensure that the subject matter is communicated in an engaging and meaningful way, this thesis involves the development of visually appealing and easy-to-understand learning content. In addition, an accompanying quiz will allow learners to actively engage with the created educational materials, which should contribute to a better understanding and enhanced retention of knowledge.

2 Foundations

2.1 Privacy

Although the notion of privacy has existed for centuries, its legal recognition did not come until the 19th century. With the invention of instantaneous photography and the rise of gossip in newspapers and magazines, the private sphere of people was increasingly put at risk [3]. Samuel Warren and Louis Brandeis were among the first scholars to recognize the necessity of protecting the privacy of individuals from the complex challenges introduced by these new developments. As a result, they published the legal article "The Right to Privacy" [4] in the Harvard Law Review to raise awareness of this issue. This article defined privacy as the "right to be let alone" [4], which includes legal protection from being subjected to emotional distress by publishing private information. This seminal article formed the basis for our understanding of privacy today and also contributed significantly to its legal recognition. [3]

The protection of personal information has become of even more significant concern since then. Especially with the advent of the Internet, privacy has become one of the most influential challenges we face daily. As personal data has proven highly profitable, companies are collecting vast amounts of it to extract valuable insights and add significant business value to their organizations. However, utilizing individuals' personal information without explicit consent violates their privacy and infringes on their right to informational self-determination. Therefore, it is imperative to protect sensitive data from unauthorized access. To accomplish this, one must first be able to define privacy.

In the last 130 years, there have been several attempts to find a common definition for the term privacy [3]. In addition to the "right to be let alone" [4], it has also been described as the right "to be protected against intrusion" [5] and the "control of personal information" [3]. Although all of these definitions may seem reasonable at first glance, one will quickly realize that there is no single correct definition, as it heavily depends on the context of a particular situation. Since a detailed exploration of the concept of privacy would be highly complex and beyond the scope of this thesis, we will not expand on this further. When we refer to privacy in the remainder of this work, we will do so in the context of legal frameworks or technical measures.

Finally, the question arises as to why we should be concerned about privacy violations. Failure to protect privacy can result in serious adverse consequences that should not be underestimated. These include compromised personal security, identity theft, loss of data ownership, human profiling, or even physical danger [6]. Given these considerations, protecting privacy is not only essential but indispensable. Although there are many approaches to achieving this protection, adopting robust laws and regulations is critical in preserving the right to privacy.

2.2 Laws & Regulations

As discussed in the last section, privacy is paramount, and many threats could jeopardize it. Thus, several laws and regulations have been established to protect individuals from privacy infringements. This ensures that organizations do not exploit sensitive data for monetary or malicious purposes without the data owner's consent. Below, we discuss three of the most prominent privacy laws that this thesis will refer to several times.

GDPR The General Data Protection Regulation (GDPR) is a regulation in the European Union (EU) aimed at protecting the personal information of data subjects[1] [8]. To ensure that sensitive data is not mishandled or exploited by a malicious organization, the GDPR regulates the entire lifecycle of such data. This includes data at rest, in transit, or in use. Furthermore, it grants individuals residing in the EU a number of rights, such as the *right to access*, the *right to erasure*, and the *right to data portability*. Any organization around the globe is held responsible for ensuring that each EU citizen's rights are upheld and must therefore take various measures to comply with this regulation. Non-compliance with the GDPR can result in heavy fines and should therefore be avoided at all costs. Due to its comprehensive scope and its effectiveness in ensuring privacy protection, it is considered a significant milestone in the efforts to prevent privacy infringements.

CCPA The California Consumer Privacy Act (CCPA) is a law aimed at protecting the privacy of citizens who reside in the state of California. It is a seminal statute that gives individuals the right to make informed decisions about their personal information. Furthermore, it is considered the most stringent privacy legislation in the U.S., as it establishes several new rights, including the *right to know*, the *right to opt-out*, and the *right to restrict*. [9]

The CCPA only applies to large organizations with a high gross annual revenue ($\geq$ \$25 million) that collect data from a large number of California residents ($\geq$ 100.000). As with the GDPR, severe penalties are imposed if these organizations fail to comply. [9]

HIPAA The Health Insurance Portability and Accountability Act (HIPAA) is a regulation in the U.S. designed to protect sensitive patient information in the healthcare industry. Every *covered entity* handling medical data, which among others include healthcare providers and clearinghouses, is subject to this federal law. Its goal is to protect all U.S. citizens from the misuse or mishandling of their medical information. To achieve this, strict regulations are imposed not only on the use of this sensitive data but also on its disclosure. This empowers individuals to make their own decisions regarding the use and dissemination of their information. Non-compliance with HIPAA can result not only in financial penalties but also in criminal charges. [10]

2.3 Privacy-Enhancing Technologies

A large number of laws and regulations have been enacted to preserve the privacy of individuals. This benefits data owners tremendously, as it allows them to have more control over their personal information. Nevertheless, a large number of organizations still rely on the collection and processing of sensitive data in order to obtain valuable business insights. Thus, the question arises as to how useful information can be derived from data without compromising the privacy of the data subjects. To bridge this gap, Privacy-Enhancing Technologies (PETs)

[1] The term data subject refers to "an identified or identifiable living individual to whom personal data relates" [7].

have been developed to allow organizations to work with sensitive information while still adhering to various data protection laws and regulations.

Privacy-Enhancing Technologies can be defined as "technologies that embody fundamental data protection principles by minimising personal data use, maximising data security, and/or empowering individuals" [11]. Nowadays, a wide range of these PETs are available to provide a technical solution for protecting sensitive data. Each PET has unique capabilities and can be used for a variety of use cases, possibly even allowing for innovations and advancements in various industries. Among the best-known PETs are Federated Learning and Differential Privacy. These are just some of the PETs that are currently experiencing an upswing in research and have already found application in the industry.

Although Privacy-Enhancing Technologies can provide a high level of security, widespread adoption is far from being achieved. This is because major challenges must be overcome if these technologies are to be deployed in an organization. As discussed in the introduction, these include technical complexity, lack of incentive, scarcity of experts, and the absence of widely accepted standards. Addressing these issues is imperative to ensure that Privacy-Enhancing Technologies become ubiquitous.

In addition, while PETs offer many benefits and can positively impact numerous organizations, we would like to emphasize that they are not a panacea for all privacy-related challenges. Each PET has its advantages, challenges, and unique capabilities. However, given that there are a large number of highly comprehensive privacy laws, it is impossible for PETs to be all-encompassing. Throughout this work, we will explore several Privacy-Enhancing Technologies and explain the specific use cases for which each can be used. First, however, it is crucial to know how to convey such a topic to a non-technical audience effectively. Therefore, in the following section, we will look at the basics of creating learning content.

2.4 Learning Frameworks

As mentioned in this thesis's introduction, this work aims to introduce a non-technical audience to the most predominant Privacy-Enhancing Technologies for data processing and analysis. To achieve this, it is essential to provide appropriate learning content about these PETs tailored to the intended readership. Many teaching and learning frameworks offer guidance to support the successful development of learning content. In this section, we would like to discuss two of them: Bloom's Revised Taxonomy and Gagné's Nine Events of Instruction.

2.4.1 Bloom's Revised Taxonomy

Learning objectives are statements that "describe the knowledge, skills and attitudes that students are meant to have acquired by the end of an educational component" [12]. The importance of Learning Objectives should not be underestimated. They explain to the student what they must do and how well they must perform in order to successfully complete the learning content [13]. Therefore, Learning Objectives facilitate students' learning by giving them a clearer picture of what is expected of them.

Bloom's Revised Taxonomy is a learning framework that can help define and articulate Learning Objectives for educational purposes. It is structured in two dimensions, the Knowledge Dimension, and the Cognitive Process Dimension. An illustration of Bloom's Revised Taxonomy is shown in Figure 2.1. [14]

The Knowledge Dimension establishes four different types of knowledge, spanning from concrete to abstract. By defining the type of knowledge to be gained, the student is informed about what kind of knowledge they are expected to acquire. [14]

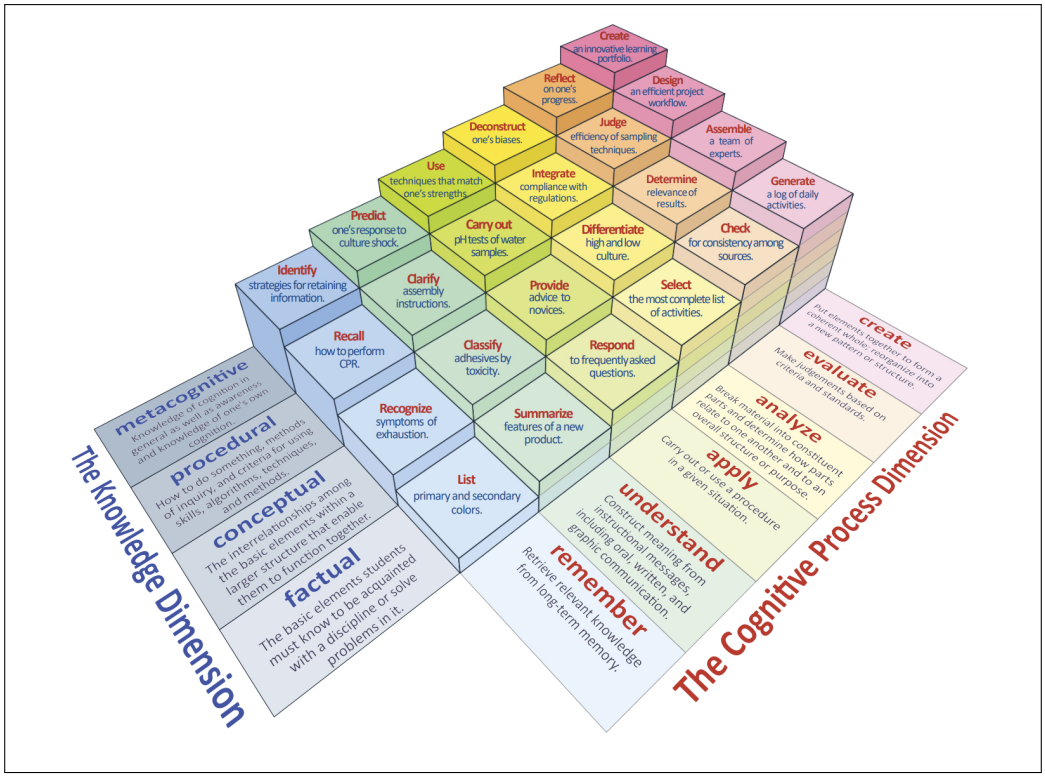


Figure 2.1: Bloom's Revised Taxonomy (Source: [14])

The Cognitive Process Dimension describes a continuum of thinking skills that range from more basic to more advanced levels. Generally, lower-order thinking skills are included and form the basis for higher-order thinking skills. [14]

By defining the type of knowledge to be acquired and the type of thinking skills needed, appropriate Learning Objectives can be established. These can then guide students in their learning process and also facilitate the creation of focused, engaging, and effective learning content.

2.4.2 Gagné's Nine Events of Instruction

Gagné's Nine Events of Instruction provide a theoretical learning framework designed to help educators develop effective teaching materials. These events attempt to provide guidance on how to create an optimal learning environment by addressing the conditions needed for effective learning. Gagné's Nine Events of Instruction can be combined with Bloom's Revised Taxonomy to increase the effectiveness of the created learning content and improve students' learning experience. [15]

The Gagné's Nine Events of Instruction include [15]:

> Gain attention of the students, Inform students of the objectives, Stimulate recall of prior learning, Present the content, Provide learning guidance, Elicit performance, Provide feedback, Assess performance, and Enhance retention & transfer.

3 Related Work

As already stated in the introduction, there is a large amount of literature on Privacy-Enhancing Technologies. However, many of these publications tend to be complex and intended for an expert readership. As the lack of education in this area has become apparent in recent years, there have already been multiple papers that attempt to make this topic more accessible to a wide audience. Hereinafter, we will discuss related works to this thesis and elaborate on exactly how our thesis fills a gap in current literature.

On the one hand, a large amount of literature already focuses on providing an overview of a wide range of Privacy-Enhancing Technologies. Nonetheless, these publications are often not written for a non-technical audience or cover the concepts of each PET only briefly. As a result, the information provided is often either too overly complex or insufficient to obtain a comprehensive understanding.

On the other hand, some publications focus exclusively on a single PET and are tailored to a non-technical audience. However, these papers are generally rather extensive and can overwhelm a non-expert readership. Thus, they might not be particularly helpful for an audience that only wants an overview of the most prominent PETs.

This thesis aims to fill this gap in current research by providing a comprehensive introduction to the most relevant PETs that a non-technical audience can easily understand. We would like to note that this section does not provide an exhaustive list of related works. Considering the vast body of literature, we could only focus on the publications that we believe are most representative of the field.

3.1 PET Overviews

First, we would like to turn our attention to publications that do not limit themselves to a single technology but rather provide an overview of numerous Privacy-Enhancing Technologies.

The sharing of data has significant benefits in most industries. In the financial sector, in particular, data sharing would be of great significance as it would allow for better risk management, combating transaction fraud, and providing personalized services. However, there is an inherent trade-off between data utility and privacy, as simply sharing customers' raw data may infringe on their right to privacy. Therefore, there are many laws and regulations that govern the handling of sensitive information. This restricts many financial institutions and prevents them from reaping the benefits of data sharing. The paper by Blake et al. aims to address this issue by providing an overview of the main PETs that can be leveraged for secure and private data sharing in the financial sector. The paper starts by addressing the benefits and challenges of data sharing between financial institutions and then moves on to explain the fundamental concepts of five Privacy-Enhancing Technologies. Finally, the paper demonstrates their data-sharing capabilities by exploring several potential use cases of these PETs for financial institutions. This publication was developed primarily for executives to give them a better grasp of the great capabilities of these PETs. However, the entire paper is also suitable for a non-technical audience, as the PETs are explained in a simplified manner and are illustrated with numerous examples. Unlike our work, however, it does not provide a comprehensive overview of each PET, covering only the most fundamental concepts of each

technology. So, although it serves as an excellent introduction to the topic, it is insufficient to provide someone with a comprehensive and nuanced overview of these Privacy-Enhancing Technologies. [16]

Another noteworthy publication in the area of Privacy-Enhancing Technologies is the report "From privacy to partnership: The role of Privacy-Enhancing Technologies in data governance and collaborative analysis" that was developed by The Royal Society in collaboration with the Alan Turing Institute. The paper's objective is to explore how PETs can contribute to higher privacy guarantees and improved collaborative data analysis. It was also developed with a broad audience in mind, including stakeholders from industry, government, and the academic field. Therefore, much of the content is also relevant to a non-technical audience. The report covers various topics to achieve its goal, including the emerging PET marketplace, standards, assessments, and assurances. In addition, numerous use cases are presented, and a set of strategic recommendations are provided to facilitate further advancements in this field. However, instead of focusing on the specific technical explanation of each PET, the report focuses primarily on the role and impact of these PETs on data governance and collaborative data analysis. This alternative perspective on the subject can be considered excellent complementary literature to this thesis. [17]

National statistical offices (NSOs) provide governments with data that is both of high quality and of great relevance. The information they collect is highly valuable, as it can facilitate informed decision-making. However, most public institutions, private companies, and individuals are reluctant to share their sensitive information because they fear that this data could be breached by a cyber-attack or even just by published statistics. The question arises whether there is a way to share and process data without compromising the privacy of the entities that contributed their sensitive information. A comprehensive report by the United Nations seeks to answer this question by providing a guide on Privacy-Enhancing Technologies that can be leveraged to create official statistics in a secure and privacy-friendly manner. First, the paper elaborates on the general motivations behind the adoption of PETs and also lists some of the inherent obstacles one might face. Then, various aspects of the most relevant PETs are discussed, including their concepts, security models, drawbacks, and use cases. The paper concludes with a discussion of standards and regulatory issues. While this guide is primarily aimed at NSOs, it can also help many other stakeholders, such as policymakers and corporate lawyers, who want to understand the privacy implications of these technologies. In line with our work, the goal of this guide is to provide an overview of a wide variety of PETs. To achieve this effectively, it also presents several highly illustrative use cases to highlight the potential applications. However, in contrast to our thesis, some of the content and explanations in this guide are unsuitable for a non-technical audience. In addition, a large amount of content (standards, regulations, etc.) is only of interest to a very specific target group. So, although the report contains information that is also helpful to a non-technical audience, it should rather be used for its intended purpose - as a guide for National Statistical Offices. [18]

The paper by C. Adams provides an introduction and overview of the most relevant Privacy-Enhancing Technologies. As a lecturer at the University of Ottawa, he identified the need for introductory literature in this area that is suitable for a non-expert audience. Hence, the intended audience of this paper are graduate and upper-year undergraduate students in computer science and software engineering. However, Adams notes that the book can also be helpful to other groups, such as researchers and practitioners. First, this paper discusses the problem that although many Privacy-Enhancing Technologies exist, it is difficult to obtain an overview of what technologies are currently available. This results in the problem that many people wish to utilize PETs but are unaware of which is best suited for their needs. To solve this problem, Adams creates a classification of PETs that allows for comparisons and contrasts

between them and provides a way to identify which is best suited for a particular use case. To accomplish that, the author categorizes the Privacy-Enhancing Technologies through the use of a privacy tree. Subsequently, he discusses the concepts, strengths, and weaknesses of the most relevant PETs, such as Federated Learning and Differential Privacy. However, due to technical jargon and certain background knowledge requirements, it is not accessible to a non-technical audience. So, the paper by C. Adams recognized the need for educational materials but only provided them for a technical audience. In contrast, our work aims to provide an overview of PETs to an audience with no prior knowledge in this field that wishes to gain a comprehensive overview of the leading Privacy-Enhancing Technologies. [6]

3.2 PETs

As we pointed out in the introduction to this chapter, some publications focus on one PET and attempt to make it more accessible to a broad audience. Although we cannot provide a comprehensive list of these publications, we would like to examine one paper for each PET that falls into this category.

3.2.1 Federated Learning

There is very little practical guidance for organizations on implementing and deploying Federated Learning (FL) systems. This can often make the adoption of Federated Learning a lengthy and costly endeavor. The report "Federated learning: an introduction – Considerations and practical guidance for prospective adopters" was created by the Open Data Institute after extensive research in the field of Federated Learning. Its objective is to address this issue by providing guidance in order to help organizations make informed decisions about adopting Federated Learning and to assist them in exploring and testing this technology. Thus their target audience are various stakeholders within an organization. To achieve this goal, the paper discusses several important aspects of FL, including its definition, capabilities, limitations, and maturity. In addition, the document provides guidance on the governance and deployment of FL systems and includes various case studies. Even though its content is primarily intended for organizations, the paper makes the topic also understandable for a non-technical audience. However, it only focuses on a single Privacy-Enhancing Technology rather than providing a general overview of the major technologies in this field. Thus, it is not suitable for individuals seeking a survey of the most relevant PETs. Moreover, instead of solely focusing on the concepts, benefits, challenges, and applications, it also provides practical guidance for organizations, which is beyond the scope of our work. [19]

3.2.2 Differential Privacy

Understanding and implementing Differential Privacy (DP) can be complex, but it also has a wide range of crucial applications in various industries. Therefore, Wood et al. have created a paper with the aim of introducing the concept of Differential Privacy and its data protection guarantees to a non-technical audience. This is intended not only as a guide for organizations processing the data but also to educate data owners about the level of privacy protection they can expect from DP. In particular, the paper discusses DP's possibilities and limitations in ensuring the privacy of sensitive data, some practical considerations one has to bear in mind when implementing it, and some tools to perform a differentially private analysis. In order to make this content more accessible to a non-technical audience, they use several illustrations and examples. In addition, the paper presents a large number of real-world applications to

demonstrate its practical utility. Similar to our work, Wood et al. identified the need to make a complex Privacy-Enhancing Technology accessible to a non-expert audience. However, as the paper only focuses on one Privacy-Enhancing Technology, it is not able to provide its audience with guidance on what PET is best suited for different scenarios. But in contrast to our work, this allows Wood et al. to discuss Differential Privacy in greater depth and thereby discuss certain aspects of this PET that are beyond the scope of this thesis. [20]

3.2.3 Homomorphic Encryption

Another PET that has high potential, especially in the future, is Homomorphic Encryption (HE). The concept of HE has already existed for more than 40 years. Since then, much research has been devoted to making HE more comprehensive and efficient. The paper by Acar et al. is a survey that captures the major developments and advances of Homomorphic Encryption schemes. It introduces the concepts and the different types of Homomorphic Encryption along with an overview of existing implementations of Gentry-type Fully Homomorphic Encryption schemes. In addition, potential avenues for future research are discussed. However, we would like to note that, for the most part, this survey was not carried out with a non-technical audience in mind. On the contrary, the target audience are scholars and practitioners who seek to thoroughly comprehend and deploy HE systems. As such, only its discussion of the fundamental concepts is suitable for a non-technical audience. However, in contrast to our work, it is much more comprehensive and hence also deals with the technical and mathematical aspects of HE. [21]

3.2.4 Secure Multi-Party Computation

Secure Multi-Party Computation (SMPC) is another complex PET that has received significant attention in recent years. The paper by Lindell achieves to provide an introductory overview of the concepts of SMPC. To this end, the requirements for an SMPC system, its security guarantees, and the threat model are presented in an understandable way. In addition, already implemented real-world applications of SMPC are included to illustrate the wide range of possible use cases. More complex and not ideally suited for a non-technical audience is the description of the basic cryptographic primitives such as Shamir Secret Sharing. In addition, unlike this thesis, Lindell's paper is not explicitly intended for a non-technical audience, as it still requires some background knowledge in order to understand the paper in its entirety. However, it is important to note that much of its content is still accessible to a non-technical readership. Thus, even for individuals that don't have a background in computer science, Lindell's paper can still serve as a valuable source to grasp the foundations of SMPC. [22]

3.2.5 Zero-Knowledge Proofs

Zero-Knowledge Proofs (ZKPs) are another promising PET that can be used to ensure the privacy of sensitive data. However, its widespread adoption is not yet within reach, as the current lack of standardization is still an obstacle that needs to be overcome. To tackle this problem, several experts have joined forces to develop the "ZKProof Community Reference". The objective of this reference is to provide guidance on how to implement ZKP systems successfully. It covers an explanation of the basic concepts, techniques, and gives recommendations for adopting ZKPs. In addition, it includes a comprehensive overview of possible use cases. Although much of the reference is intended for a technical audience only, it is important to the creators of this work that some of the content is also accessible to a wider audience so that not only professionals but also the general public can gain more insight

into this particular PET. Therefore, they use descriptive examples to more effectively convey the theoretical knowledge. Since the reference deals exclusively with ZKPs, it provides a much more detailed explanation of this technology than our thesis. Furthermore, the purpose of our work is to communicate only the fundamental concepts, benefits, challenges, and applications of a wide range of PETs, so no information regarding the implementation or deployment of these systems is included. Therefore, when actually developing these systems, one should rather consult the community reference to get a more comprehensive overview of this technology. [23]

4 Methodology

As discussed in the introduction, this thesis aims to provide educational resources to a non-technical audience. This chapter describes the various methodologies used to achieve this goal.

First, three research questions are presented to define the scope of this thesis. We then explain the quantitative analysis we used to select the PETs discussed in this thesis. Next, a Systematic Literature Review was conducted to gain a fundamental understanding of each technology. Finally, the methodology used to develop the educational materials will be explained and analyzed.

4.1 Research Questions

The following research questions were formulated to be answered within the scope of this thesis:

1. What are the most prevalent PETs for data processing and analysis?
2. What are the characteristics, benefits, challenges, and applications of the selected PETs?
3. How can one convey this information in a meaningful and engaging manner to ensure a better understanding of the selected PETs?

Since the purpose of this work is to be an introduction and overview of the topic, not all PETs can be discussed within the scope of this thesis. Therefore, the first research question seeks to determine the most relevant PETs for data processing and analysis, which will then be discussed in later chapters.

The purpose of the second research question is to provide the reader with a fundamental understanding of each identified PET. By discussing the characteristics, benefits, and challenges associated with each technology, we can distinguish the use cases of the different PETs and select the right one for a particular scenario. In addition, practical applications and examples of these technologies will further demonstrate their relevance in the real world.

The goal of the final research question is not to gather or summarize new information but to convey the material from the previous research questions in an engaging and meaningful way. Providing educational resources to a non-technical audience is the main objective of this thesis. Therefore, it is not only the content that is important but also the way in which it will be communicated. By utilizing different teaching methods, we aim to actively engage the reader and ensure a good understanding of key concepts.

4.2 Quantitative analysis

The goal of the quantitative analysis is to select the most relevant PETs for data processing and analysis to ensure that the audience gains a broad and useful overview of the subject. Therefore, this is the first and crucial step in writing this thesis.

We started by defining an objective metric that we can use to decide whether one PET is more relevant than another. In this analysis, a PET is more *predominant* if it has more publications in the current literature. We recognize that this metric is not an indicator of the effectiveness or practical utility of a given Privacy-Enhancing Technology. However, we are confident that this approach is sufficient for the purposes of this work, as it still allows us to identify the most researched PETs in the current literature.

4.2.1 Process

First, the scope of the quantitative analysis had to be defined. To ensure credibility and relevance, the analysis focused only on conference papers, journals, and research articles published as of 2013. In addition, we also selected four academic databases to include in the analysis, namely **IEEE Xplore**, **ACM Digital Library**, **Scopus**, and **ScienceDirect**.

Next, it was necessary to determine a list of PETs to be included in the quantitative analysis. For this purpose, we used a literature review conducted by Garrido et al. [24], which provides a comprehensive list of PETs that can be used for data processing. From this list, a pre-selection was made to exclude obsolete PETs or technologies that were not relevant to this thesis. This resulted in the final enumeration of PETs to be included in the quantitative analysis (see Table 4.1).

PET Name	**Alternative Notation**
Differential Privacy	
Homomorphic Encryption	
Secure Multi-Party Computation	Secure Multiparty Computation
Zero-Knowledge Proofs	Zero-Knowledge Proof
Federated Learning	
Trusted Execution Environments	Trusted Execution Environment
Privacy-Preserving Data Mining	
Private Information Retrieval	
K-Anonymity	
L-Diversity	
T-Closeness	
Pseudonymization	Pseudonymisation

Table 4.1: Quantitative Analysis - PETs and their Notations

Now that we had identified the PETs to be included in the analysis, it was necessary to determine their predominance in the current literature. This was achieved by searching for the specific PETs in the selected databases. Included in this analysis were all papers that contained the name of the chosen PETs in the title, in the abstract, or in the author keywords. Given that some of these technologies also have slightly different notations, these were also included in the searches. These different notations included plural forms, hyphens, and differences in American and British English. Table 4.1 shows the various notations for each PET that were also included in this analysis. Possible acronyms of these PETs were not considered as we encountered a large number of identical acronyms that had different meanings. This would have led to a distortion of the findings.

Finally, the results for each PET from each database were documented and summarized. It should be noted that the data collection was conducted on the 24th of March, 2023. Since all digital libraries included in this analysis continually expand their database by adding new papers, results at other points in time will most likely differ from the results of this analysis, albeit only slightly.

4.2.2 Data Deduplication

Digital libraries contain publications that are not exclusive to their database, which could lead to potential overlaps. This could skew the results of this analysis. To deal with this problem, the data was deduplicated by first exporting the search results to a CSV or BibTex format and then importing each record into Python.

However, downloading all search results was an issue in the ACM Digital Library and ScienceDirect, which limited the number of downloaded references to 50 and 100 records, respectively. When the number of published articles exceeded these limits, it was necessary to export these search results into multiple tables and then concatenate them into one data set.

Furthermore, data sets provided by the different databases used different formatting, and IEEE Xplore had a minor issue with properly escaping characters. Thus, the data had to be first manually prepared and sanitized in order to be useful for the final step of the analysis.

Lastly, the tables were merged, and the records were deduplicated by title. The Python code used for the deduplication can be seen in Listing 4.1. The final results of the analysis were then structured, ranked, and visualized in a bar chart. Only the five most predominant PETs were chosen to be included in this paper to avoid covering too many or irrelevant technologies.

```
# Load CSV-Files
acm_data = pd.read_csv("rsc/ACM_Digital_Library/x.csv", sep=",", encoding="UTF-8")
ieee_data = pd.read_csv("rsc/IEEE_Xplore/x.csv", sep=",", encoding="UTF-8")
scienceDirect_data = pd.read_csv("rsc/ScienceDirect/x.csv", sep=",", encoding="UTF-8")
scopus_data = pd.read_csv("rsc/SCOPUS/x.csv", sep=",", encoding="UTF-8")

# Rename columns
acm_data.rename(columns={'Title': 'Title'}, inplace=True)
ieee_data.rename(columns={'Document Title': 'Title'}, inplace=True)
scienceDirect_data.rename(columns={'Title': 'Title'}, inplace=True)
scopus_data.rename(columns={'Titles': 'Title'}, inplace=True)

# Select only the title
acm_titles = acm_data["Title"]
ieee_titles = ieee_data["Title"]
scienceDirect_titles = scienceDirect_data["Title"]
scopus_titles = scopus_data["Title"]

# Merge tables
merged_data = pd.concat([acm_titles, ieee_titles, scienceDirect_titles,
    scopus_titles], axis=0)
merged_data.reset_index(drop=True, inplace=True)

# Deduplicate
deduplicated_data = merged_data.drop_duplicates(keep='first')
deduplicated_data.reset_index(drop=True, inplace=True)
```

Listing 4.1: Python Code for Data Deduplication

4.3 Systematic Literature Review

This thesis aims to provide a foundational understanding of the most prominent Privacy-Enhancing Technologies. To accomplish this, we heavily rely on literature that discusses these technologies' concepts, theories, and practical applications. Intending to identify the most relevant publications from this large body of work, we conducted a Systematic Literature Review (SLR) based on Kitchenham [25].

4.3.1 Research Questions

A Systematic Literature Review is insufficient to answer all the research questions we want to address in this thesis. Therefore, we first had to determine which research questions could be answered by conducting this SLR. The quantitative analysis already answered the question of which PETs are the most prevalent in current literature. In addition, an SLR fails to answer the research question of how to facilitate the development of engaging learning content. With all of this in mind, we concluded that the SLR can only address the second research question:

What are the characteristics, benefits, challenges, and applications of the selected PETs?

4.3.2 Search Strategy

Now that we have defined the objectives of this SLR, we would like to elaborate on its methodology. First, we want to discuss the search strategy that helped us identify the most relevant papers.

A simple search for the names of the identified PETs was usually insufficient, as instead of information on characteristics or use cases, this resulted in mainly finding papers that explain a specific implementation. Therefore, we refined our search strategy by employing automated searches using search strings derived from the research question. The following search strings helped us immensely to find the most relevant papers on a wide variety of PETs:

- "[PET name]"
- ("[PET name]") AND (overview OR introduction OR basics OR fundamentals OR concepts OR principles)
- ("[PET name]") AND (applications OR benefits OR challenges)
- ("[PET name]") AND (survey)

The *[PET name]* is a placeholder for the names of each PET and their various notations. Similar to the quantitative analysis, these different notations include plurals, hyphens, and differences between American and British English. Table 4.1 shows the various notations we included in our search. Databases searched for publications included **IEEE Xplore**, **Scopus**, and the **ACM Digital Library**.

To ensure that the papers we find are relevant and impactful, we focused only on papers published within the last ten years and sorted the search results by the number of citations. However, since it is impossible to include all literature meeting these criteria, we only considered the ten most frequently cited publications for each search string. Additional inclusion and exclusion criteria are listed below:

- *Inclusion criteria*:
  - Papers have to be relevant and readily used in current literature.

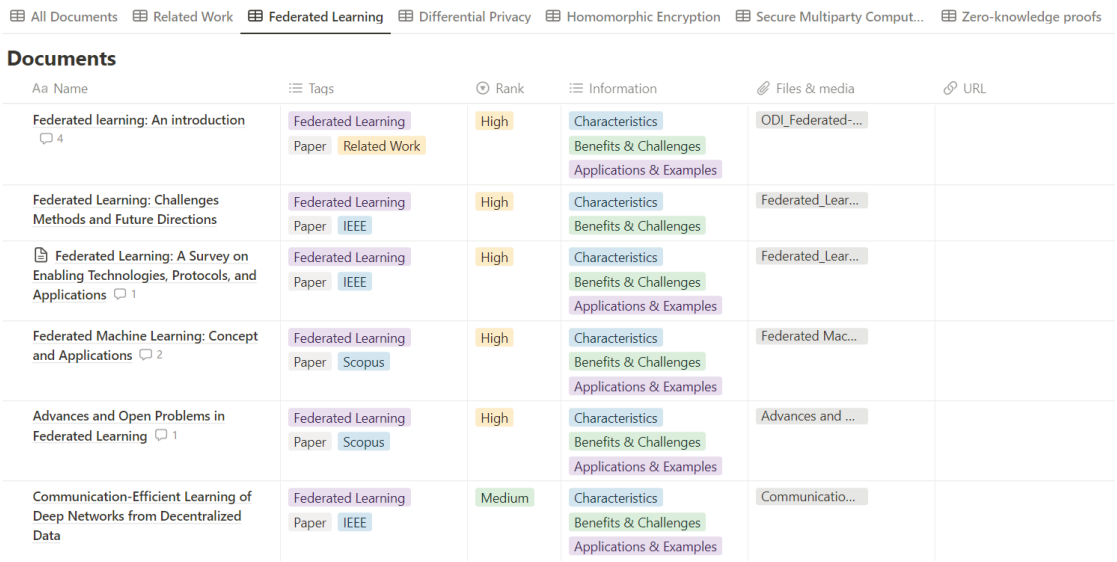


Figure 4.1: Notion Database

  - Papers have to address the characteristics, benefits, challenges, and applications of the selected PETs in the context of data processing and analysis.

- *Exclusion criteria*:
  - Papers that focus on legal requirements or policies.
  - Papers that talk in detail about a specific implementation or use case of a PET.
  - Papers that provide no useful information to a non-technical audience.

For the most part, this search strategy allowed us to retrieve the relevant publications from a large body of literature. However, for some Privacy-Enhancing Technologies, there is insufficient introductory literature, especially for a non-technical audience. Therefore, we also had to manually search gray literature to supplement the knowledge obtained from the SLR. Additional sources used for this thesis include magazines, websites, and blog posts. Although we primarily attempted to reference highly renowned papers, in the case of lesser-known PETs, we had to resort to citing gray literature.

4.3.3 Data Extraction & Synthesis

After selecting the papers to include in our work, we had to extract their relevant information. Since we were confronted with a large body of literature, a structured approach was essential in ensuring that data extraction and synthesis were comprehensive and effective.

This was accomplished by capturing the publications in a central database using the Notion web application. In this database, papers were categorized by PET and sorted according to their relevance to our work. Then the collected papers had to be analyzed by highlighting their most critical information with the help of color coding. This information was grouped into three different groups/color schemes: **Definition & Characteristics** (Blue), **Benefits & Challenges** (Green), and **Applications & Examples** (Purple). Furthermore, we created an additional column to the database, which indicates which paper contains what type of information. All these efforts allowed us to access the collected data in an efficient and systematic way.

Figure 4.1 shows the result of this process, which helped us tremendously in drafting a large part of this thesis and synthesizing the most relevant information from this SLR.

4.4 Development of the learning content

The development of learning content for a non-technical audience is at the heart of this thesis. In the last chapters, we discussed the motivation behind its development. Now we would like to discuss how we achieved this thesis' objectives by designing effective, engaging, and easy-to-understand educational materials. This includes the approach we took to develop the Learning Objectives, the individual Learning Paths, and the corresponding Quiz for assessment.

4.4.1 Learning Objectives

As the developed learning content is intended for a non-technical audience, we are unable to cover every Privacy-Enhancing Technology extensively. Therefore, before we can articulate the Learning Objectives, we must first use Bloom's Revised Taxonomy [14] (see section 2.4.1) to determine the categories of knowledge and the cognitive processes we would like to focus on.

Florian Messmer's bachelor thesis [26] shows how Learning Objectives in the context of Privacy-Enhancing Technologies can be mapped to the different categories of Bloom's Taxonomy. The results of his research are presented in Table 4.2.

	Remember	**Understand**	**Apply**	**Analyze**	**Evaluate**	**Create**
Factual Knowledge	I want to know what different PETs exist.	I want to know the various use cases for PETs.	I want to be able to follow discussions about PETs.	I want to be able to differentiate PETs.	I want to verify statements about the features of PETs.	I want to be able to classify a new PET.
Conceptual Knowledge	I want to know how and why PETs work.	I want to know how PETs are integrated into a system architecture.	I want to be able to create my own architectures involving PETs.	I want to be able to compare PETs based on their principal attributes.	I want to decide on which PET would be most suitable in a given system environment.	I want to create metamodels for PETs.
Procedural Knowledge	I want to identify use cases for applying PETs.	I want to be able to explain how different PETs are implemented.	I want to be able to implement PETs in a system environment.	I want to compare different ways to implement PETs.	I want to decide on the best way to implement a PET in a given situation.	I want to contribute to the development of new PETs.
Metacognitive Knowledge	I want to know the limitations of PETs.	I want to identify the limitations of a given PET implementation.	I want to be able to give strategies for optimizing the implementation of PETs.	I want to compare PET implementations based on their effectiveness.	I want to evaluate PET implementations and develop recommendations.	I want to find new use cases to which PETs could be applied.

Table 4.2: PET Learning Objectives Mapped to Bloom's Revised Taxonomy (Source: [26])

With the help of this table, we were now able to select the specific categories of Bloom's Taxonomy that are appropriate for a non-technical audience and that can guide us in developing the learning content. Table 4.3 presents these identified categories.

In the course of Messmer's work, a number of legal professionals were asked which of the Learning Objectives in Table 4.2 they considered essential for their role-specific needs. Table 4.4 shows the result of the conducted survey, with the color in each cell indicating whether the majority of legal experts consider the corresponding Learning Objective essential to their role. We would like to emphasize that the Learning Objectives they identified align almost

	Remember	Understand	Apply	Analyze	Evaluate	Create
Factual						
Conceptual						
Procedural						
Metacognitive						

Table 4.3: Bloom's Revised Taxonomy - Categories Covered in This Thesis

	Remember	Understand	Apply	Analyze	Evaluate	Create
Factual						
Conceptual						
Procedural						
Metacognitive						

Table 4.4: Role Specific Needs of Legal Experts (Source: [26])

entirely with the categories of Bloom's Taxonomy that we decided to focus on. As legal experts do not have any technical background and can thus be considered a non-technical audience, this supports our argument that the identified Learning Objectives are indeed suitable for a non-technical readership. [26]

Based on these results, it was now possible to formulate the Learning Objectives. First, it was essential to limit the number of Learning Objectives to six to avoid the students being confronted with overwhelming detail and, thus, a lack of focus [13]. With the help of further guidance of [12] and [14] on how to formulate them, we articulated the following Learning Objectives for our learning content:

The student is able to

- list the most predominant PETs in the current literature.
- explain the features and characteristics of the selected PETs.
- describe the benefits and challenges of the selected PETs.
- describe the applications and use cases of the selected PETs.
- distinguish between PETs and determine which of them is best suited for a particular problem.
- state two successful implementations for each PET and an example where a privacy breach occurred that could have been prevented by a successful PET implementation.

4.4.2 Learning Paths

After having developed the Learning Objectives we want to achieve with our learning content, we would now like to turn our attention to how we created each Learning Path. As we discussed before, we aim to provide a comprehensive overview of the five most relevant PETs in current literature, tailored to a non-technical audience. Thus, in the scope of this thesis, we developed one Learning Path for each Privacy-Enhancing Technology. Each Learning Path consists of three Learning Nuggets: **Definition & Characteristics**, **Benefits & Challenges**, and **Applications & Examples**. Together, these Learning Nuggets are designed to meet all the Learning Objectives we strive to achieve with our educational materials.

Since we have already described how we gathered the necessary information through a

Systematic Literature Review, we now want to discuss how we used the collected data to create engaging, visually appealing, and easy-to-understand learning content.

First, it was necessary to define the target audience of the learning content. As thoroughly discussed throughout this thesis, the objective is to enable individuals from diverse backgrounds to understand better the characteristics, benefits, challenges, and potential use cases of the leading Privacy-Enhancing Technologies. Therefore, we directed our focus to a non-technical audience without prior knowledge. This required us to explain technical terms in detail, simplify complex concepts, and provide relatable examples in order to make the topic more accessible and help readers understand PETs in a more comprehensible way.

After determining the target audience, it was apparent that they would require explicit instructions on what they needed to do and how well they needed to perform in order to complete the learning content successfully [15]. To accomplish this, we decided to provide learners with detailed Learning Objectives prior to each of the Learning Nuggets and Quizzes.

Gagné's Nine Events of Instruction

According to Gagné's Nine Events of Instruction [15], it is also critical to capture students' attention so that they become more engaged and invested in the learning process. We accomplished this by introducing each chapter with an overarching example from the healthcare sector, particularly focusing on a general hospital in the United States. This example is then used to describe a challenge faced by a hospital and how a particular PET can help address it. This should pique readers' interest, as they will be immediately exposed to a real-world example of how PETs can be applied to ensure privacy. Although we did not want to use a real hospital in this example, we were willing to use the fictitious name "SEBIS General Hospital" (SGH) for improved clarity and understanding.

In addition, [15] and [27] suggested that students should be presented with both examples and non-examples. Therefore, we decided to present not only two successful implementations of each PET but also an example where a privacy breach occurred that could have been prevented. This should give them not only a clear picture of how these technologies can already be applied in practice but also what can happen if privacy is neglected.

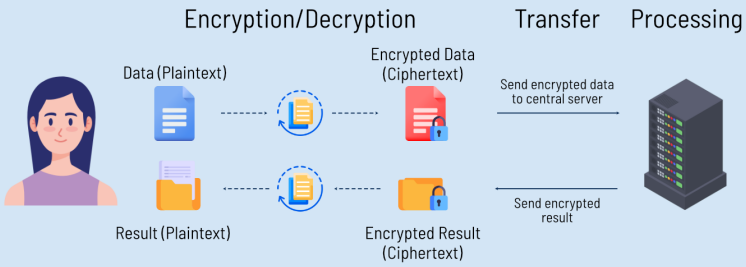
Furthermore, it is the instructor's responsibility to provide students with learning guidance in order to help them learn more effectively [15]. This was achieved by giving real-world examples and making extensive use of visualizations to convey the information more lucidly (see Figure 4.2). When creating these examples and visualizations, we consulted the paper "E-learning methodologies and good practices" [27], which also gave us further guidance on how to develop engaging text and graphics. All of our various strategies helped reduce the complexity of abstract concepts and made the content more accessible to a non-technical audience.

Another essential aspect of effective learning content is the ability of the students to elicit their performance [15]. This can be achieved through the design of effective quizzes. We also developed solutions for these quizzes to provide the learners with feedback on their performance. See the following subsection for more details on how we created them.

The methods listed above for effectively delivering learning content are only a summary of the various strategies we used to create it. That is why we would also like to refer to [14], [15], and [27], which played an integral role in the further development of the learning content.

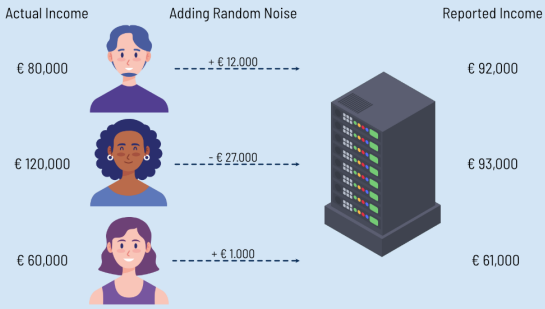
Homomorphic Encryption

With traditional encryption methods, the encrypted data is protected when **transmitted over a network** or **stored in a database**. However, if you want to process the data and gain valuable insights about it, you have to decrypt it first, which can lead to serious privacy issues. Homomorphic Encryption is an encryption scheme that can solve this problem, **as the data remains encrypted even during processing**. This means that a third party not authorized to read the data can still perform useful operations on the ciphertext. These operations on the ciphertext reflect a corresponding operation on the plaintext. As a result, after the ciphertext has been processed by an external party, it can still be sent to an authorized person, who then decrypts the ciphertext again and thus obtains the result of the operation. **This outcome is then identical to the result that would have been obtained if the operations had been performed on the plaintext instead.** At no point was it necessary to disclose the raw data to the unauthorized entity actually processing it since the ciphertext does not reveal any information about the plaintext without the decryption key.



Noise

Differential Privacy can be accomplished by adding random noise to the data to protect the specific information of the individual participants.



Example: Researchers want to determine the **average household income** in Germany. John Doe is asked to participate in this study, which gives the researchers access to his personal data. However, he fears this information could become public and is therefore reluctant to share his data. Since many people share Joe's fear of privacy breaches, researchers leverage Differential Privacy. Thus, anyone is allowed to add random noise to their answer. In this case, they can choose a random number between ***-$30,000*** and ***+$30,000*** and add it to their actual income before reporting it to the researchers.

Figure 4.2: Visualizations of Privacy-Enhancing Technologies

4.4.3 Quiz

As we have already explained in the last section, it is crucial to elicit students' performance, as this can give them an indication of the areas in which they still need to improve [15]. It can also help instructors by providing feedback on how effective the learning content they are creating is and how they might need to change their strategies to improve the learning experience for students. Therefore, with the creation of individual Learning Paths, we now have the task of developing a corresponding quiz that tests the learners on the knowledge imparted to them. In this section, we will briefly discuss how we designed it.

With a non-technical audience in mind, we developed a quiz and a corresponding solution for each of the five most predominant Privacy-Enhancing Technologies, with each quiz containing a variety of exercises. However, we first had to determine what content needed to be included and what methods we could use to test their knowledge. To achieve this, we had to utilize our already developed Learning Objectives in conjunction with Bloom's Revised Taxonomy to guide us in designing and developing the quiz.

Next, we had to determine what type of questions we wanted to include in each quiz. In selecting the possible question formats, we consulted the paper "E-learning methodologies and good practices" [27], which lists and explains the different types of questions that can be incorporated into an interactive quiz.

Ultimately, we decided to develop a "True-or-False" and a "Multiple Responses" exercise for each PET. In a "True-or-False" quiz, the participant must decide whether a particular statement about a PET is either true or false. However, it is not necessary to provide an explanation for their answer. In the "Multiple Responses" exercise, the participant must answer questions by selecting one or more possible answers. Similar to the "True-or-False" quiz, no explanation is required here. When applicable, we also developed an "Ordering" exercise in which one has to arrange the steps of a particular PET process chronologically. Figure 4.3 shows examples of each question format. [27]

For further advice on creating an effective quiz, we have drawn on the paper "E-learning methodologies and good practices" [27]. One key recommendation emphasized that all incorrect choices must be plausible to make learning challenging and engaging. These wrong options should also not be designed to distract learners but to reflect common mistakes instead. Furthermore, in the "Multiple Responses" exercise, all choices should be approximately equal in length to not give any clues as to which answer is correct. Care must also be taken to ensure that questions are not ambiguous, as this could confuse participants and cause them to lose interest in the learning process. [27]

Access to Learning Content

We have now covered the most critical aspects of the methodology we used to create the learning content. Unfortunately, due to its massive volume, we cannot include the created educational material in this thesis. However, all learning content, as well as the quiz and the corresponding solutions, can be downloaded as a PDF file at this link:

https://1drv.ms/f/s!Ag9Y6xObvis1ry2mvE58uj05X8fj?e=Scj1uX

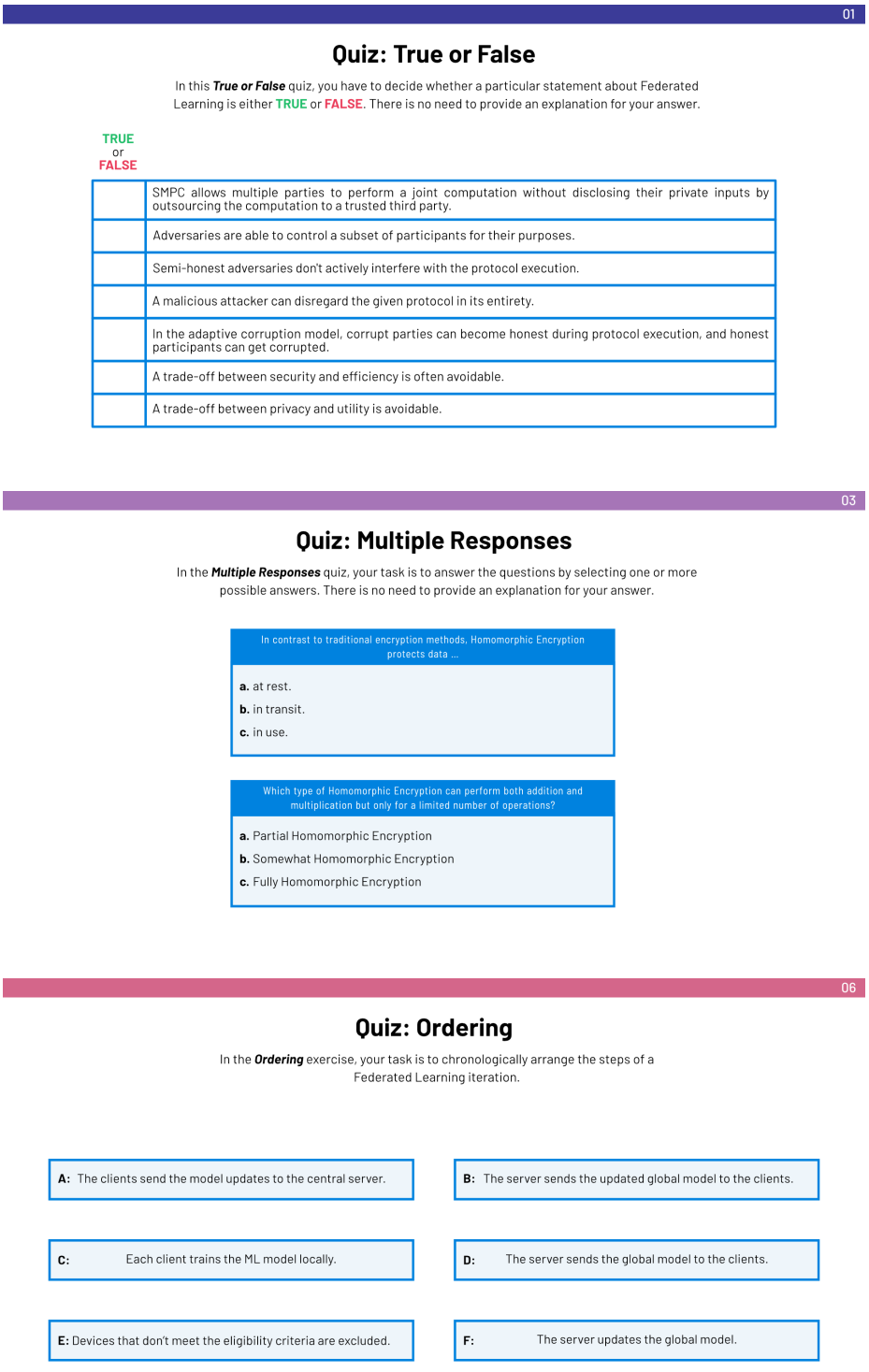


Figure 4.3: Examples of Question Formats - True or False, Multiple Responses, and Ordering Exercises

The educational material and the accompanying quiz are provided in the form of slides created using the vector graphics program *Affinity Designer*[1] (version 1.10). All icons included in the Learning Paths are hosted by the *Freepik Company*[2], and the purchased licenses for these icons can be found at the link shown above.

Although evaluating the developed learning content is beyond the scope of this work, we would like to emphasize that validation and assessment of this educational material is still needed and can be considered an imperative task for future work.

[1] Affinity Designer can be downloaded at the following link: `https://affinity.serif.com/en-us/designer/`
[2] The icons hosted by Freepik can be downloaded here: `https://www.flaticon.com/`

5 Privacy-Enhancing Technologies

This chapter aims to provide a comprehensive introduction to the most prevalent Privacy-Enhancing Technologies used in data processing and analysis. Therefore, this chapter can be considered the core of this thesis and the groundwork for the subsequent creation of the learning material.

After discussing the results of the quantitative analysis and thus identifying the Privacy-Enhancing Technologies to be addressed in this thesis, a separate section is devoted to each of the selected PETs. To improve comprehension and readability, each section follows the same structure, which is explained hereafter.

First, the definition and characteristics of these PETs are explained, already supported by illustrative examples. This should help the reader gain a basic understanding of how each technology works. To highlight the capabilities and limitations of these technologies, an overview of the key benefits and challenges is then provided. Finally, applications and real-world examples are presented to show the PETs' practicality. These examples include both successful applications in practice and examples where such technologies were not used, resulting ultimately in privacy infringements that could have been prevented. We decided to include this part to show the feasibility of PETs and to demonstrate what can happen when organizations don't use them and thus fail to protect the privacy of their customers.

Moreover, we will provide an overarching example to introduce each technology. A recurring theme throughout the thesis can help a non-technical audience better understand the practical benefits of applying PETs.

Since healthcare is one of the industries where privacy is a major concern, the adoption of PETs in this area (e.g., in hospitals or medical centers) offers many benefits [28, 29]. Therefore, we decided that the overarching example should be one from the healthcare sector, specifically a general hospital in the United States.

Although we do not want to use a real hospital in this example, we are willing to use the fictitious name *SEBIS General Hospital* (SGH) for improved clarity and understanding. At the beginning of the following sections, this example is used to describe a challenge faced by this hospital and how a particular PET can help address it.

5.1 Quantitative analysis: Results

By conducting the quantitative analysis described in section 4.2, we are able to identify the most predominant PETs in the current literature. The in-depth results are presented in Table 5.1.

It shows the number of papers found for each PET in each selected database. The columns correspond to the respective academic databases used to collect the information. The rows correspond to the individual PETs selected for this analysis. Moreover, both the sums and the total deduplicated number of papers for each PET were formed across the selected databases.

Table 5.1: Quantitative Analysis - Results

	IEEE Xplore	ACM DL	Scopus	ScienceDirect	Total	Total (Deduplicated)
Federated Learning	3729	435	6204	446	10814	7256
Differential Privacy	1,942	644	4903	374	7863	5971
Homomorphic encryption	1,842	327	4992	322	7483	5802
Secure Multi-party Computation	495	206	1936	107	2744	2275
Zero-knowledge proofs	414	108	1485	83	2090	1677
Trusted execution environments	412	216	943	43	1614	1260
Privacy-Preserving Data Mining	237	37	744	71	1089	853
Private Information Retrieval	371	45	666	22	1104	812
L-Diversity	101	13	388	29	531	431
Pseudonymization	48	9	241	162	460	402
T-Closeness	52	4	138	36	230	186

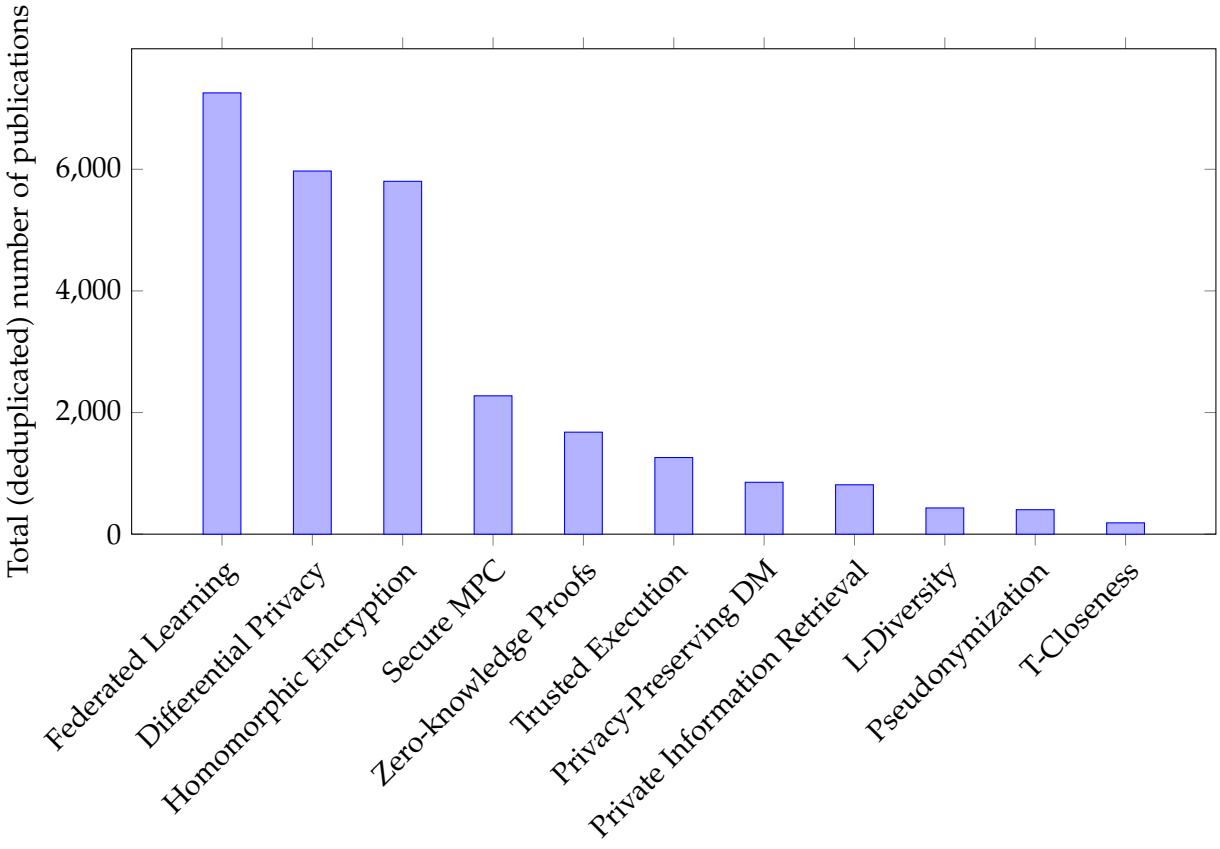


Figure 5.1: Total (Deduplicated) Publications per Technology

Figure 5.1 visualizes the total deduplicated number of publications for each technology in a bar chart. It is evident that some technologies are significantly more prevalent in the literature than others. Since we cannot cover all PETs in the field of data analysis and data processing, this thesis only deals with the most important Privacy-Enhancing Technologies. Therefore, we only included the five most predominant ones that emerged from this analysis:

1. **Federated Learning**
2. **Differential Privacy**
3. **Homomorphic Encryption**
4. **Secure Multi-Party Computation**
5. **Zero-Knowledge Proofs**

5.2 Federated Learning

Before we look at the definition and characteristics of Federated Learning (FL), let us first consider the following problem statement:

The SEBIS General Hospital (SGH) would like to be able to predict COVID-19 in its patients at an early stage. To predict this illness, they can develop a machine learning (ML) model[1] that identifies the disease with a high probability. However, such models require a large data set to ensure the desired accuracy. Since SGH does not have enough patient data, they would like to collaborate with five other hospitals that have exactly the same goal. In traditional machine learning, the data from all six hospitals would be stored in a central location, and then a central server would train the ML model using this data. However, in this case, the hospitals have to share their patients' sensitive information. This is often not possible due to various regulations, such as HIPAA, that attempt to prevent privacy violations. Therefore, developing such a prediction system would not be possible. [31, 32]

Federated Learning can now be used to overcome this problem. Instead of collecting and processing the data centrally, the client SEBIS General Hospital can use Federated Learning to train their ML model. The term *client* will be used to refer to all entities that have sensitive information - data that should not be exposed - and that participate in the FL process. This may include individual devices or even entire organizations [33]. The goal of this section is to provide the reader with a fundamental understanding of Federated Learning by discussing its definition, characteristics, and categorizations.

5.2.1 Definition & Characteristics

Federated Learning allows ML models to be trained without storing the raw data in a central location. This is achieved by shifting the responsibility of training the model from a central server to the clients. By providing each client with an ML model, the training process can be performed locally without them having to disclose any of their sensitive information. As a result, their raw data remains local at its point of origin and does not need to be revealed to an external party. Instead, the clients only need to transmit updates to a shared global ML model in order to improve the accuracy of its future predictions. The global model, a centralized model of the clients' aggregated results, resides on a central server responsible for coordinating the FL process between the different devices. [19, 34, 32, 35, 36]

Cross-Device vs Cross-Silo Federated Learning

It is necessary to distinguish between two different types of FL before discussing its implementation. Federated Learning can involve the use of multiple edge devices to train a model, such as smartphones, laptops, and smart wearables. This is called Cross-Device Federated Learning, and the number of participating devices can potentially reach millions. In contrast, Cross-Silo Federated Learning relies on data residing not on individual devices but across multiple organizations that hold data silos[2] and want to collaborate. Cross-Silo FL can be used by companies to cooperate in order to achieve a common goal. The number of participants is

[1] A machine learning (ML) model is a type of algorithm that can be leveraged to "recognize patterns in data or make predictions" [30]. In order to achieve accurate results, the model must be fed data that allows it to learn and identify underlying relationships. This process is called training the ML model, and the data that is used to accomplish this is referred to as training data. By providing an ML model with a large amount of this data, the model's predictions can be improved until it is able to perform tasks with an acceptable level of accuracy.

[2] A data silo is an isolated database owned by a specific group or entity that contains information that should not be shared [37].

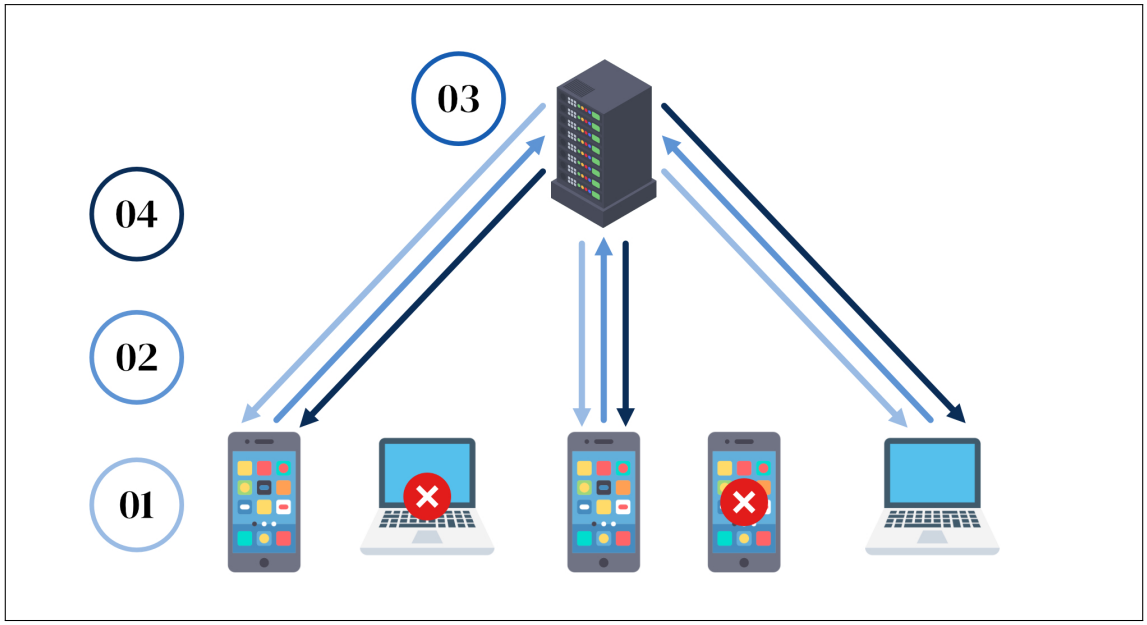


Figure 5.2: Federated Learning Process

typically much smaller compared to Cross-Device training. Still, Cross-Silo FL can be even more complex because the collaborating organizations need to align their goals and objectives. Although both types of FL are important, this thesis will focus on Cross-Device FL, as there is more research and already successful applications in this area. [19, 35]

Federated Learning Process

A common approach to implementing FL according to McMahan et al. [34] is shown in Figure 5.2, which is described again below [19, 35]:

1. First, the central server must determine the clients that can participate in the FL process based on pre-established eligibility criteria. Once the available clients have been identified, the server can send them all of the information they need to train the ML model locally.

2. Each client trains the ML model with its local data to determine a model update that can be used to improve the global model. The clients then send this update to the central server.

3. After the central server receives the updates from the clients, they are aggregated, evaluated, and used to update the global model to improve its accuracy.

4. This new global model is then transmitted to the clients that took part in this iteration.

Since FL is an iterative process, the steps mentioned above can be repeated with the updated model until the learning process is complete [35].

As can be seen in Figure 5.2, in the first step, only a number of clients are selected based on previously set eligibility criteria. The question arises as to why we can only use a subset of the devices when we would benefit greatly from a larger amount of data. The rationale behind this is that we need to ensure that the users of these devices do not suffer any negative consequences from participating in the FL process. These potential drawbacks could include loss of battery or computing power and even financial losses. To counteract this, devices should only be selected in an iteration when they are idle, charged, attached to a power supply,

and have a free internet connection [34, 35]. However, one should keep in mind that these are only guidelines and that these inclusion criteria have to be evaluated on a case-by-case basis.

Furthermore, we discussed that the clients don't have to expose their sensitive information. Now the question arises if no raw data is sent to a central server, what kind of information is transmitted in the first place? So far, we have chosen to call them *updates* without explaining what that entails.

First, we need to know what objective we want to reach with Federated Learning. In FL, we want to accomplish a goal with a specified accuracy. To achieve this, we must optimize the ML process and thus minimize the predefined loss function [34]. A loss function calculates the difference between the predictions of our ML model and the true outcome [38]. The higher the difference, the worse our model is. Thus, if our ML model is perfect, the loss function would be 0.

In the context of Federated Learning, the updates sent by each client contain information on how to improve the accuracy of the global ML model in order to optimize its predictions and minimize the loss function. This can be made possible, for instance, by transmitting *model gradients* to the central server [34]. Since we only want to introduce Federated Learning here, we will not go into detail about what this actually entails.

Categories of Federated Learning

With the reader having a basic understanding of how FL works, we would now like to divide FL into three categories based on the feature and sample ID space: Horizontal Federated Learning, Vertical Federated Learning, and Federated Transfer Learning [39]

Before discussing the differences between these types, we must first explain the feature and sample ID space by providing an example in the financial sector.

The feature space corresponds to the attributes that are collected from clients [36]. For banks that collect information about their customers, this could include age, gender, income, address, and credit score.

On the other hand, a sample ID corresponds to the unique identifier of a data point [39]. The sample ID space thus refers to the collection of identifiers that uniquely identify each data point collected by a client. For instance, a bank in the US will have a vastly different sample ID space than a bank in Germany, as the customers are most likely not involved with both banks at the same time. [36]

Horizontal Federated Learning: Horizontal Federated Learning (HFL) can be applied when the feature space is the same but the sample ID space is different. For example, two banks, A and B, are located in a completely different country. Bank A and B want to collaborate to develop an ML model without disclosing sensitive information about their customers. Since the two banks are very similar, they collect the same features about their customers, such as age, income, and credit score. However, their customer bases differ greatly since they operate in different locations. Therefore, the feature space is the same, but the sample ID space differs. [36]

Vertical Federated Learning: Vertical Federated Learning (VFL) can be used when the feature space is different, but the sample ID space is similar. In this case, Bank A wants to cooperate with an insurance company located in the same city to develop a joint ML model. Since the insurance company does not collect the same attributes as the bank but instead collects information about the payment status of bills due and claims history, among other things, they do not have the same feature space. Since both are located close to each other, they almost certainly overlap in sample ID space, i.e., customers who

go to both the bank and the insurance company. Now Vertical FL can be applied to develop an ML model. [36]

Federated Transfer Learning: The question arises as to how to proceed when both the feature and sample ID space differ. [39] In this case, Federated Transfer Learning (FTL) can be employed. For example, Bank A wants to collaborate with an insurance company located in a completely different region to develop a joint ML model. Since both work in different industries and are in completely different locations, the feature space and the sample ID space differ significantly. As a result, neither HFL nor VFL can be used for training the ML model. However, FTL can now be applied to enable collaboration between the two companies. [36]

Depending on the real-world use cases, one of these types of FL has to be selected for implementation. This decision affects - among other things - the architecture and security concerns of the corresponding Federated Learning system [39]. Since the various implementations for the different types of FL are too extensive to cover in this thesis, this will not be discussed further. However, if interested, we can refer to the work of Yang et al. [39], where the subject is explained in more detail.

For the sake of completeness, it should be mentioned that Federated Learning does not need to be applied in isolation but is often used in conjunction with other PETs such as Differential Privacy, Homomorphic Encryption, and Secure Multi-Party Computation. The relationship between these technologies and Federated Learning will be discussed in later sections.

5.2.2 Benefits & Challenges

Having explained the concept of Federated Learning and its characteristics in the previous section, we would now like to deal in more detail with the possibilities and limitations of this PET. We will first start with a short overview of the advantages before we go into detail about the challenges that have to be considered when implementing FL.

Benefits of Federated Learning

Privacy & Security The first benefit we would like to point out is that Federated Learning safeguards sensitive information and, therefore, the privacy of its participants. By collecting only the updates needed to improve the FL model, the raw data never leaves the participants' devices. This mitigates the problems that come with centralized data management and unauthorized data access. [32]

Compared to traditional machine learning, this also improves data security with respect to data sharing [19]. With Federated Learning, malicious attackers do not even get the opportunity to harvest the raw data on the way to a central server. However, it is important not to fall into a false sense of security. Privacy and security issues remain even when Federated Learning is employed [19]. Their occurrence and the question of how to mitigate them are discussed in section 5.2.2.

Legal & Regulatory Compliance Protecting customer data is not only ethically imperative but also mandated by laws and regulations. As explained in section 2.2, there are several regulations, such as the GDPR, CCPA, and HIPAA, that organizations must comply with to avoid harsh penalties. Although Federated Learning cannot be considered a comprehensive solution for these regulations, it can certainly be used to be compliant with portions of the mandated laws and regulations. [19]

Improved Model Performance & Predictions So far, we have only discussed how Federated Learning can enhance privacy and security. However, training an ML model using Federated Learning itself has many advantages.

With Federated Learning's privacy-friendly features, it's possible to create much larger data sets, as clients who were partially constrained by laws and regulations now have a way to work around them. With their data well protected, more potential customers may even be willing to participate in the FL process. This could result in more data points and, thus, greater model performance and accuracy. [19, 40]

It should also be mentioned that FL has some advantages in terms of predictions when compared to the traditional machine learning approach. In Cross-Device Federated Learning, each device has a local ML model. This enables real-time predictions since there is no significant time difference between the emergence of new data and the use of this data to train the model. In addition, it is possible to make offline predictions, as no Internet connection is required. [32]

Cost Reduction & Scalability The adoption of Federated Learning in an organization can lead to cost reductions, as a large amount of resources can be saved compared to traditional ML [19]. One of the cost savings can be achieved by the clients not having to send their raw data to a central location but only a small update. Thus bandwidth and internet traffic can be reduced [19]. Also, much less hardware is required since instead of a central server, the task of computationally intensive training is delegated to the participating clients [32, 40]. This also leads to better scalability, as adding new clients does not automatically result in higher training times [19]. In addition, it is generally very easy to add new devices to the network once the FL system is deployed [19].

Challenges of Federated Learning

Although FL brings many benefits, there are also currently many obstacles that must be overcome, which are discussed hereafter.

High Communication Cost In Federated Learning, communication between the central server and the clients plays a central role. In each iteration, many messages are exchanged between them. The ML model to be trained can become very complex and even consist of millions of parameters, resulting in a large amount of information being transferred back and forth each time [40]. As a result, although no raw data is sent to a central server, communication costs still play a major role. In particular, Cross-Device Federated Learning involves a large number of devices [32]. Since these devices usually have very low bandwidth and an unreliable network connection, communication can not only be costly but also a bottleneck in the FL process [33].

To address this problem, one can both reduce the size of the messages to be sent and the number of communication rounds [32]. One way to achieve this is by compressing the model or the model gradients in such a way that the data to be transmitted is reduced considerably [40]. However, one must be careful not to compress the model too much, as this transformation degrades the FL model. Therefore, finding a balance between data quality and data compression is crucial. In contrast, Importance-based Updating is a strategy to reduce the number of communication rounds while still creating an accurate model [40]. By not simply sending all updates to the central server but only those that are relevant and significantly impact the model, you can reduce the communication overhead considerably. As Luping et al. [41] showed with their implementation of "Communication-Mitigated Federated Learning"

(CMFL), Importance-based Updating can reduce the number of communication rounds, while even achieving slightly higher accuracy than conventional FL algorithms.

Device Heterogeneity A Federated Learning process usually involves a large number of devices, all of which contribute to training a global ML model. However, these devices are not identical; on the contrary, they differ in several characteristics [33]. The problem of heterogeneity in these devices is discussed in more detail in this paragraph.

The devices participating in the FL process differ in their hardware (e.g., CPU, memory, etc.), network connectivity, and battery power. This results in different computing, communication, and storage performance [32, 33]. These differences can have a large impact on the FL process, as each iteration can only be performed as fast as the slowest device. As a result, large delays can be caused by only a few clients. This is also known as the Straggler effect and can be considered a bottleneck for Federated Learning [40].

Furthermore, only a fraction of the total devices participate, as many do not meet the desired eligibility criteria. Since the devices are also subject to changing conditions (e.g., connection limitations), it also often happens that devices suddenly stop participating midway through the FL process [32]. It is therefore critical to be prepared for abrupt dropouts by implementing an FL system that can tolerate failures [33]. Considering all these arguments, Aledhari et al. [32] propose that all developed FL systems should satisfy the following three conditions:

- Anticipation of low amounts of participation
- Tolerance of different device hardware
- Handling of failed devices in network settings

A brief overview of how to address some of the issues arising from device heterogeneity can be found in the paper by Li et al. [33]

Data Heterogeneity In addition to device heterogeneity, there is data heterogeneity in FL processes, as the data collected from customers have distinct properties and are usually far from perfect for training an ML model. For example, this data is often unbalanced, which can lead to poorly trained models [32, 34]. Data is unbalanced when the number of data points varies significantly from one client to another. One of the problems that arises is that clients with more data points are favored while clients with less data are treated unfairly [33]. This can make it more challenging to build a global model and thus decrease its prediction accuracy.

Data heterogeneity can lead to major problems in training an ML model [33]. However, there are several algorithms that can handle this type of heterogeneity, making Federated Learning applicable even when the underlying training data is imperfect [34].

Privacy Threats Although privacy is one of the benefits of Federated Learning, there are still potential threats that could compromise the privacy of participants. Instead of raw data, model updates are sent to a central server. However, these model updates can still be used to draw conclusions about the participants. When a malicious third party tries to infer client properties through model updates, it can be called an inference attack [19, 33]. We want to focus on the two most common types, the membership attack and the reconstruction attack.

The goal of the membership attack is to find out if a specific data point exists in a data set and has been used for training the ML model. To accomplish this, a malicious third party typically creates an attack model that has the sole goal of predicting whether a particular data

point has been used for training. To effectively train this attack model, the attacker needs a data set that originates from the same or a similar source. However, if the attacker is successful, the consequences for the clients' privacy are catastrophic. For example, if a medical study is being conducted to determine the outcome of a particular disease, one could determine which individuals are in the data set and thus have that specific disease. [42]

On the other hand, the reconstruction attack attempts to learn more about the training data itself. As covered in the last section, gradients are sent to the central server for updates in each FL iteration. However, if one is not cautious, these gradients can still expose sensitive information about the client. [42]

As one can see, although Federated Learning is a Privacy-Enhancing Technology, there are still some privacy concerns to watch out for. However, it should be noted that these risks can be mitigated by combining Federated Learning with other PETs, such as Differential Privacy or Homomorphic Encryption [19]. But having more privacy usually comes at a sacrifice in terms of resources, performance, and accuracy [19, 33].

Security Threats Finally, there are certain security concerns that you need to be aware of. First, the inference attacks mentioned in the previous paragraph are not only a privacy issue but also a security concern that should be monitored [19]. However, in this section, we would like to focus on other types, namely Poisoning and Free Riding attacks. [40, 42]

In Poisoning Attacks, a malicious third party attempts to negatively affect the model, i.e., reduce the accuracy of the model [19]. One way this is achieved is through Data Poisoning. In Data Poisoning, the clients' raw data sets are manipulated in such a way that the model is degraded [40]. Another way to negatively impact accuracy is the Model Poisoning attack. In this attack, it is not the client data that is altered but the updated model that is sent to the central server to improve the global model [19]. Especially in large FL systems with many participants, Model Poisoning is even more efficient than Data Poisoning because the actual model can be directly influenced [40]. There is already some research on how to mitigate these kinds of attacks. If interested, a comprehensive overview can be found in the paper by Lim et al. [40].

The Free-Riding Attack has a completely different objective. Instead of contributing to the global model, the attacker does not want to participate in the FL process but still wants to profit from it. This is achieved, in part, by the free rider pretending to have very few data points and, therefore, not being able to train as much. This saves resources such as computing power. However, the free rider still gains access to the global model and thus reaps all the benefits. [40]

We want to take the opportunity to state that this is not a comprehensive list of attacks that can occur. However, this is a complex and extensive subject that cannot be covered further within the scope of this work.

5.2.3 Applications & Examples

Having established a basic understanding of this PET, let us now explore where Federated Learning can be used in practice. Through the examples in the last sections, some applications should already be familiar. However, in the following, we will briefly summarize where Federated Learning can be applied.

Areas of Application

Organizations As we have already shown with the SEBIS General Hospital example, organizations can also act as clients within an FL system. This allows companies to collaborate with other companies to create a shared ML model that benefits all parties involved. That this does not work with traditional ML is often due to the fact that the individual companies do not want to share their private data or are not even allowed to do so because of various privacy laws [33]. Therefore, Federated Learning could gain a lot of traction, especially in the healthcare and finance sector, in which collaboration is highly beneficial but usually not possible due to regulatory restrictions [19].

Smartphones In addition, Federated Learning is also commonly used in smartphones to build a global ML model across a large number of devices. Features such as next-word prediction can greatly improve the user experience but require a lot of personal information that users do not want to disclose. In this case, FL can be used to provide users with intelligent applications without compromising their privacy. [33]

IoT Devices Nowadays, Internet of Things (IoT) networks are becoming increasingly ubiquitous. Autonomous vehicles, fitness trackers, or even smart appliances contain sensors and an internet connection, allowing them to collect and exchange information in a real-world setting [33]. However, the collected data is mostly privacy sensitive and should not be exposed. For instance, a smartwatch can collect information such as blood pressure, number of steps, etc., and give the user advice based on that information to improve their overall health. However, this type of health information should not be at risk of being compromised. By leveraging Federated Learning, this can be achieved without jeopardizing the users' data.

Ideal Use Cases

Now, however, the question remains what problems should be solved with Federated Learning. The paper by McMahan et al. [34], in which Federated Learning was first introduced, identifies the following characteristics as ones that make a problem ideal for the application of Federated Learning:

1. FL should be used especially when, for a particular application, training with data collected in the real world has a clear benefit compared to training with proxy data. Proxy data is only representative of real-world data but can still be used as a substitute when the actual data is not available or accessible. For example, you can create an ML model to improve mobile users' typing experience by predicting their next word or even entire sentences. There is no good proxy data because the style of chat and text messages differs significantly from texts that can be found online (e.g., websites, documents, etc.). As a result, real-world data would serve much better as training data than the available proxy data. [34]

2. In addition, FL should only be applied if the raw data contains sensitive information or if the size of the data to be transmitted is larger than the size of the ML model. As stated many times before, the raw data does not need to be sent to a central server but remains at the local clients. This not only increases privacy but also reduces the need to send large amounts of data over the network, thus saving communication costs. [34]

Primary Industries

Although Federated Learning is being used in all sorts of industries, only two sectors that are among the "most promising application area[s]" [19] according to Shteyn et al. are discussed below.

Healthcare is one of the areas that can benefit most from the widespread adoption of Federated Learning. Collaboration among multiple medical institutions could help create ML models that use medical data to determine or even predict the diagnosis of common diseases [19]. It has already been used to predict COVID-19 in patients at an early stage, but the potential application areas within the healthcare sector are vast [31].

Another industry that could benefit greatly from Federated Learning is the financial sector. Like the healthcare sector, individual banks are reluctant to share their data because of various laws and regulations. Federated Learning could be used to achieve significantly better fraud detection and potentially more accurate credit scores for their customers [16]. This would allow banks to reduce their risks and thereby save financial resources.

Maturity & Adoption

Federated Learning is a PET that has already been extensively researched. Moreover, companies such as Google have shown that the real-world adoption of FL has the potential to be very successful [43]. Nevertheless, there are currently very few successful implementations of FL, and the existing deployments are mainly in the private sector. This is due to the fact that there are still several questions that need to be addressed in the context of Federated Learning, such as ethical issues, governance, and compliance. In addition, although there is much theoretical research out there, in order to be useful for all types of organizations, this research would need to be accompanied by practical guidance. Since this currently exists only to a limited extent, the introduction of FL is often an undertaking that takes a lot of time and money. [19]

According to Gartner [44], Federated Learning will not gain widespread adoption for another 5-10 years and thus has not yet achieved its zenith. Companies that nevertheless decide to implement Federated Learning right now must expect to encounter initial difficulties. However, as FL gains widespread adoption in the industry, they will also reap the benefits of being an early adopter [19].

Although it is not within the scope of this thesis to provide guidance on the deployment of FL systems, we can refer to the paper by Shteyn et al. [19] that covers this in more detail.

Examples

A Preventable Privacy Breach First, we want to discuss an example in which a company had a privacy breach that could have been largely prevented if they had implemented Federated Learning.

Ai.type is a virtual keyboard app available on both IOS and Android. This app offered a high level of customization and used an ML model to facilitate users' writing by making typing and emoji suggestions [45, 16]. To achieve this, the company used traditional machine learning and centrally collected a variety of information from its users, including names, email addresses, contact lists, and even users' keystroke history [16]. Because they did not properly secure their central database, a privacy breach occurred in which 31 million users' data was exposed to the public [45]. If they had used Federated Learning, most sensitive information would have never left the local devices, and they still would have been able to provide all of the functionality to their customers [16].

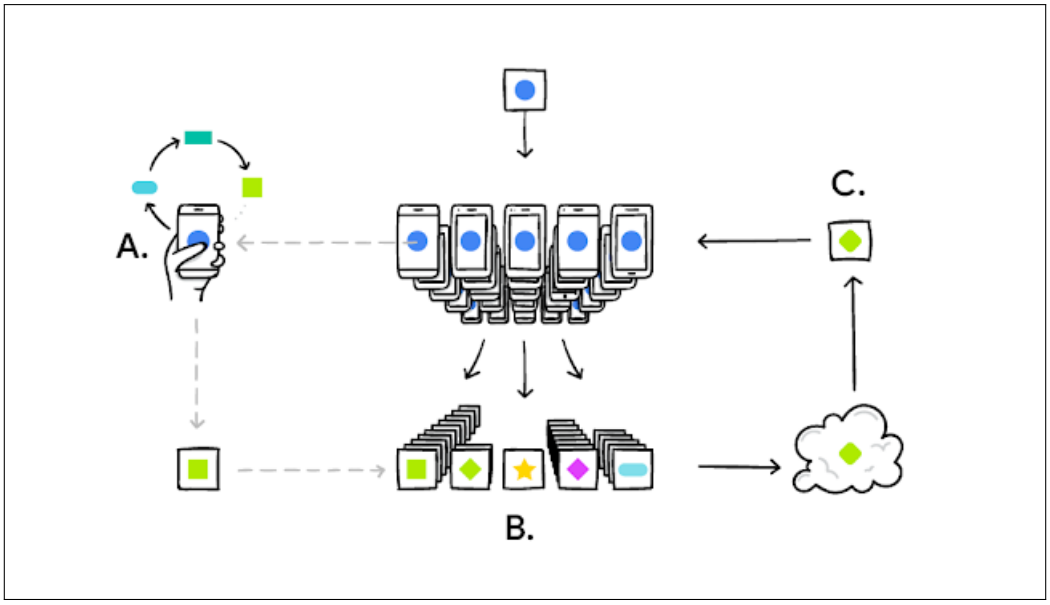


Figure 5.3: GBoard - Federated Learning Process (Source: [43])

Successful Implementations Lastly, we want to present two very well-known and successful applications of Federated Learning that show the capabilities of this PET.

The GBoard is a keyboard for mobile devices that utilizes machine learning to provide a personalized experience to its users. It offers a variety of features such as next-word prediction, word completions, and automatic corrections that allow for faster typing and an improved user experience [32]. All of these features should be provided to millions of devices while the application should remain performant and secure [19]. In addition, sensitive information about users should not be disclosed to any external party [32]. By developing Federated Learning, Google was able to achieve all of that [34]. In a blog post, they describe their FL implementation with the help of an illustration (see Figure 5.3) and an accompanying description [43]:

> "Your phone personalizes the model locally, based on your usage (A). Many users' updates are aggregated (B) to form a consensus change (C) to the shared model, after which the procedure is repeated."

GBoard's success is undeniable, as previously popular third-party keyboard apps like SwiftKey or Swype have lost their relevance as time has gone by. [19]

Apple is using Federated Learning to personalize the popular voice assistant *Siri* in order to improve the users' experience without compromising their privacy. When a user says "Hey Siri," the voice assistant automatically opens and is immediately available for questions or instructions. However, Apple wants to prevent other people from triggering this feature. Therefore, Federated Learning was leveraged to personalize Siri by building a model responsible for recognizing the user's voice. To avoid privacy violations, it is trained exclusively on local audio data on each device. However, by iteratively sending updates to a central server, the global model and, thus, its accuracy can still be improved. Nowadays, someone in a room full of iPhone users can say "Hey Siri" without launching all the other voice assistants. [46]

5.3 Differential Privacy

Let's first take another look at an issue facing the SEBIS General Hospital (SGH) before we delve into the definition and characteristics:

SGH would like to conduct a clinical trial to determine the efficacy of a novel anti-cancer treatment. After finding a group of participants, they need to conduct the experiment and collect the necessary medical data from the patients. The next step is to analyze the data to extract valuable insights and determine if the treatment was successful. However, the SGH does not have the necessary expertise to conduct such an analysis. They would like to hire an external contractor skilled in performing such an analysis on the database. However, since analyzing this data could reveal sensitive information about individual patients to an unauthorized third party, this option is not readily available to them. Violating the numerous regulations and laws that prohibit this could be very costly to SGH.

Differential Privacy can now be leveraged to help SEBIS General Hospital to accomplish this objective. By adding random noise to the output of the analysis, it can be guaranteed that no specific information about an individual participant is revealed, thus preserving their privacy [20]. By discussing the basics of Differential Privacy hereafter, we want to give the reader a basic understanding of what this entails and what needs to be considered when applying Differential Privacy.

5.3.1 Definition & Characteristics

First, it is essential to know that Differential Privacy is not a tool that can be applied directly but a rigorous mathematical definition of a privacy goal. Therefore, Differential Privacy is not an algorithm capable of providing privacy by itself but a condition that several algorithms and mechanisms can mathematically prove to satisfy. It is possible to then use such an algorithm to transform raw data so that it remains private in accordance with the definition of Differential Privacy. [47]

Since this is an introduction for a non-technical audience, we would like to first define what a query and an algorithm are before going into more detail about Differential Privacy. A query is an operation on a database. It can be used to obtain more information about the underlying data. In the case of a medical study, "How many participants are there?" would be a query that a corresponding database could process. In contrast, an algorithm is a set of instructions aimed at achieving a specific goal. For example, an algorithm can sort an arbitrary list of numbers by following specific steps. Keep in mind that this is not an explanation of Differential Privacy but of an algorithm and a query in general.

Definition of Differential Privacy

Now that we know that Differential Privacy is a mathematical definition, we would like to know what exactly it defines. Even though we do not want to discuss the actual mathematical definition, we can still give an answer to what qualifies as Differential Privacy.

Differential Privacy ensures that when an individual is removed from the data set, the result of the query fundamentally does not change [48, 49]. Thus, the conclusion of the data analysis does not change when a data point is removed from the database. As an illustrative example, consider a study in which researchers attempt to establish a relationship between the number of hours slept and income. They survey 2000 participants and find that, on average, people who sleep less have higher incomes. If one were to take an individual out of the data set, the correlation between sleep and income would, in essence, not change so that the inference would remain the same.

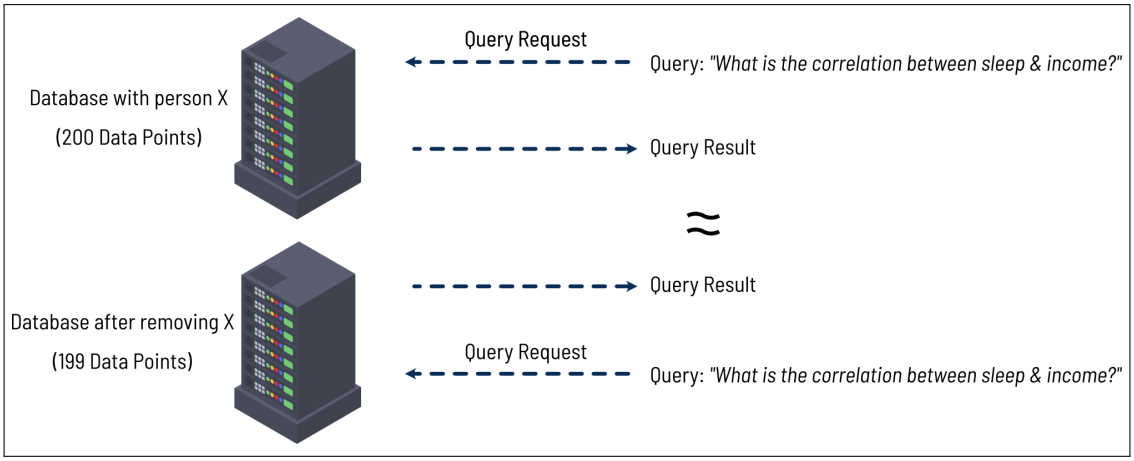


Figure 5.4: Differential Privacy's Impact on Query Results (Based on [50])

Figure 5.4 illustrates this concept by comparing the result of the same query executed on two different databases [20, 49]. The only difference between these databases is that one of them has one individual (X) removed from its data set. Since they differ in only this one data point, they are also referred to as neighboring databases [51]. If Differential Privacy is guaranteed, then the output of the two queries should essentially be the same, thus revealing no specific information about X.

Keep in mind that even if a query result does not change fundamentally, it can still produce a slightly different result [48]. How much the result can deviate is directly related to how much privacy can be ensured. This concept will be discussed later in this section in more detail.

Privacy Guarantees

This leads us to the question of what Differential Privacy can promise the participants of an analysis. It guarantees a participant that they have essentially the same privacy risk they would have had if they had never even participated at all [20, 49]. Again, we refer to the word *essential* to show that even though there is not a fundamental difference, participation still carries the risk of compromising your data [48]. However, this risk is not significantly different whether you participate in an analysis or not. The following example with Alice is inspired by Wood et al. [20], who used a comparable example to demonstrate the guarantees of Differential Privacy.

Alice is participating in a study that attempts to establish a link between the number of hours slept and income. However, she is afraid to reveal this sensitive information about herself. When the researchers analyze this correlation using Differential Privacy, Alice can be assured that no information specific to her will be disclosed. She has *essentially* the same privacy risk that she would have had if she had not participated at all.

However, Differential Privacy does not ensure that the actual results of an analysis won't put you at a disadvantage or reveal statistical information about you [47]. Dwork et al. [47] refers to this as Differential Privacy not guaranteeing "that what you believe are your secrets remain secret."

To understand what this implies, let's consider Alice again, who participated in a study to determine a correlation between the number of hours slept and income. Alice has a roommate named Bob, who knows that Alice participated in the study and that she only sleeps 5 hours per day. From this information, he concludes that she is probably wealthy, thus obtaining

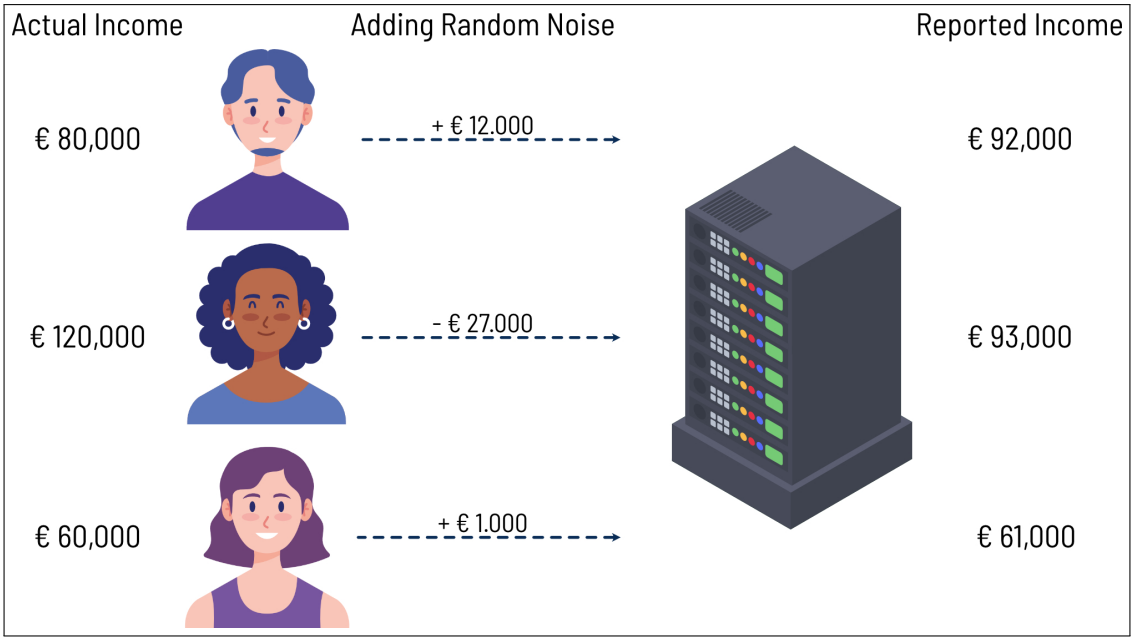


Figure 5.5: Adding Random Noise Before Data Aggregation

new private information about her financial status. The question arises whether this can be considered an invasion of privacy. Under the definition of Differential Privacy, this is not a breach. Bob could have reached the same conclusions if Alice had never participated in the study since the study's conclusion remains the same even if she is removed from the data set. The outcome of the study would be no more detrimental to Alice than if she had not participated in the study at all [52]. Bear in mind that no specific information about her was disclosed, such as her actual income [20].

To conclude, Differential Privacy ensures that no specific information about you is revealed, but it does not guarantee that the overall conclusion of the analysis cannot have an adverse effect on you. However, the latter cannot be prevented since the same conclusion is drawn whether you are in the data set or not. [20, 52]

Random Noise

After explaining what we want to accomplish with Differential Privacy, we would like to start talking about how we can actually achieve it. As explained above, Differential Privacy can be accomplished by adding random noise to the data to protect the specific information of the individual participants [20].

For example, researchers want to determine the average household income in Germany. John Doe is asked to participate in this study, which gives the researchers access to his personal data. However, he fears this information could become public and is therefore reluctant to share his data. Since many people share Joe's fear of privacy breaches, researchers leverage Differential Privacy. Thus, anyone is allowed to add random noise to their answer. In this case, they can choose a random number between -€30,000 and +€30,000 and add it to their actual income before reporting it to the researchers. An illustration of this example can be found in Figure 5.5.

Local vs Global Differential Privacy

The last example has demonstrated how noise can be introduced into the data set even before the data is aggregated. However, this is only one of two ways to insert random noise. Depending on when the noise is injected, Differential Privacy can be divided into two categories: Local and Global Differential Privacy.

In Local Differential Privacy (LDP), the data is transformed before the participants' data is aggregated [53]. The previous example, where John Doe was able to add random noise to his income before disclosing it to researchers, is an instance of LDP. The question is how the data can still be used if we add random noise to every single data point. At first glance, one might assume this would lead to poor results. However, in accordance with the law of large numbers, the more data points we include in the analysis, the more our query result approaches the real average [49]. Therefore, as the noise evens out if a large number of people participate, we can still draw meaningful conclusions from the data.

In contrast, with Global Differential Privacy (GDP), the data is aggregated by the database owner without any noise being added first [53]. The individual data points in the database are the real values and, if leaked, could lead to serious privacy infringements. Here, the noise is only added to the output of a query to allow for differentially private analysis of the data set [53]. In the case of the previous example, John Doe would share his exact income with the researchers. After gathering all the data, the researchers run a query to calculate the actual average household income of the participants. Before releasing this information, they add random noise to this result. This example shows how Global Differential Privacy can be employed to protect the participants' private information.

The question arises when to choose Local or Global Differential Privacy. The advantage of Local Differential Privacy is that one does not have to trust the database owner since the noise is added even before they receive the data. However, depending on how many people participate, LDP could also lead to worse results than GDP because every single data point is transformed, not just the output of the analysis. So if the data curator is trustworthy, GDP should be applied, as it provides better utility and, therefore, better results. Note that this is only a general guideline and that this decision should be made on a case-by-case basis. [53]

Privacy Parameter: Epsilon

Previously, we stated that Differential Privacy guarantees that the query result essentially does not change when a data point is removed from the database [48]. But even if it doesn't change substantially, the query results of two neighboring databases are not identical. This concept was illustrated in Figure 5.4. We also mentioned that the more the results of the two neighboring databases can differ, the more privacy is sacrificed [54].

The privacy parameter ϵ specifies how much the probability of a query output occurring in database A may differ from the probability of the same output occurring in database B if the same query gets executed, in terms of their ratio [54, 52]. It indicates how much privacy we are willing to sacrifice and has to be chosen by the researchers that apply Differential Privacy [54]. The smaller the value of ϵ, the more privacy can be guaranteed, and the more noise has to be injected into the data [55]. Therefore, queries that reveal more about the participants have a higher epsilon parameter than queries that extract almost no sensitive information from the data set.

One might wonder why we cannot simply set ϵ to zero to ensure that there is no difference in the results, leading to no privacy risk at all. However, this would result in the data point having absolutely no effect on the query result, so using it in our analysis would be of no benefit [55]. Thus, we couldn't learn anything useful from the data set. In the example of the

average household income study, $\epsilon = 0$ could only be satisfied by a query that always outputs a random number [56]. Since no private information was used, no sensitive data can be leaked. However, we cannot draw any conclusions about how high the average household income might be.

This example shows that choosing the *right* ϵ is a challenging task since there is an inherent trade-off between privacy and utility [52]. Thus, the value of ϵ must be decided on a case-by-case basis. This will be discussed further in section 5.3.2.

Privacy Budget

By running multiple queries on the same database, more information about an individual participant can be derived. So, the risk of your data being compromised increases with a larger number of differential analyses. However, since we want to limit the overall privacy risk, the concept of a Privacy Budget was established. It can be denoted by ϵ_{total} and is considered across multiple queries on the same data set. It is a non-negative number the data analyst sets before processing the data. Each executed query reduces ϵ_{total} by the privacy loss of ϵ_{query}. [20]

For example, a couple of researchers want to analyze a particular data set that contains private information. To protect the participants, they want to apply Differential Privacy, and based on the specific circumstances of this study, they set the Privacy Budget to $\epsilon_{\text{total}} = 1$. Now they want to run both query A ($\epsilon_{\text{a}} = 0.3$) and query B ($\epsilon_{\text{b}}$=0.4). Since queries spend the Privacy Budget, combined, they reduce ϵ_{total} by 0.7.

$$\epsilon_{\text{total\_spend}} = \epsilon_{\text{a}} + \epsilon_{\text{b}} = 0.3 + 0.4 = 0.7$$
$$\epsilon_{\text{total}} - \epsilon_{\text{total\_spend}} = 1 - 0.7 = 0.3$$

Now, the researchers wouldn't be able to execute another query with $\epsilon > 0.3$ on the same data since the Privacy Budget isn't allowed to get negative. At this point, we would like to emphasize that the Privacy Budget is not an inherent trait of Differential Privacy but a concept to illustrate that the results of multiple outputs are not independent and can be combined to find more information about the participants. [20]

Privacy Parameter: Delta

In contrast to the ϵ-parameter, the δ-parameter allows for a relaxation of the Differential Privacy constraints [48]. It specifies the probability with which the data can be inadvertently compromised [57]. Although one needs to be cautious when using this parameter, it can be a useful tool to extract more information from the given data set. As with the ϵ-parameter, choosing the correct value of δ is crucial to the Differential Privacy process [48]. If δ is set too high, the consequences can be severe. When $\delta = 0$, we refer to the corresponding algorithms as ϵ-differential private. Otherwise, they are called (ϵ, δ)-differentially private [48].

Mechanisms

Multiple algorithms can be used to inject noise and thus satisfy the condition of Differential Privacy. In the following, we only provide a very brief overview of three different mechanisms.

Laplace Mechanism: The Laplace Mechanism is an algorithm that can be employed to achieve Differential Privacy by adding noise to the output of a query based on the Laplace

distribution [57]. The amount of noise to be added depends on the selected ϵ-parameter and the sensitivity of the query [51]. The sensitivity of a query is the maximum difference between the output of two neighboring databases [51]. For example, in the case of a query that counts how many people participated in a study, the sensitivity of the query would be one, since the only difference between two neighboring databases is that one of them has one less entry. The Laplace Mechanism is very popular due to its simplicity, but it can only be used for numerical data [57].

Exponential Mechanism: Sometimes, adding noise to the data doesn't make much sense [58]. For example, if one wants to find a date on which as many attendees as possible are available, adding noise to the data and thus shifting the dates of the participants (e.g., from Monday to Tuesday) will obviously result in useless information. Sometimes it is just a matter of finding the best option from a finite list without changing the data points while satisfying the definition of Differential Privacy. In this case, the Exponential Mechanism can be employed. It can be used both for numerical and categorical query results and is, therefore, a great alternative to the Laplace Mechanism. It also takes into account both the ϵ-parameter and the sensitivity of the query. [59]

Gaussian Mechanism: The last two algorithms that we presented didn't allow for the delta parameter to be taken into consideration. That's why the Gaussian Mechanism was introduced, which works very similarly to the Laplace mechanism. This algorithm utilizes the Gaussian distribution instead of the Laplace distribution to add noise to the result of a query. However, in this case, the δ-parameter can be considered in the process, thus enabling this mechanism to handle (ϵ, δ)-differentially private analyses. [59]

This is by no means an exhaustive list of algorithms that can be used to satisfy the definition of Differential Privacy. However, it is beyond the scope of this thesis to discuss these algorithms and their individual benefits and challenges in depth. We can refer to the paper "The Algorithmic Foundations of Differential Privacy" by Dwork et al. [48] that discusses them in more detail.

5.3.2 Benefits & Challenges

Having explained the concept of Differential Privacy and its characteristics in the previous section, we would now like to discuss the benefits and challenges of this PET in more depth. After giving a brief overview of the main advantages, we will go into detail about the challenges of Differential Privacy.

Benefits

Privacy & Security First, we would like to start with the main reason why Differential Privacy is used in the first place. As discussed in detail in the last section, Differential Privacy provides strong privacy protection by intentionally adding noise to the data set. It guarantees a participant that they have essentially the same privacy risk they would have had if they had never even participated at all [20]. Thus, Differential Privacy can offer a strong privacy guarantee that can be mathematically proven to protect against various privacy attacks, such as re-identification, record linkage, and differencing attacks [20]. It is not even necessary to be able to predict the type of attacks that can occur since Differential Privacy protects against even unknown attacks or attacks that have not even been developed yet [20]. Furthermore, privacy can be assured regardless of the attacker's computational resources or knowledge about the process of our differential private analyses [50].

In addition, one must take into account auxiliary information when attempting to protect the privacy of the participants. A database cannot be considered in isolation, as nowadays, a vast amount of data is collected and easily accessible [20]. As a result, attackers can link seemingly unrelated data sets together to obtain specific information about an individual. However, if Differential Privacy is employed, any kind of linkage attacks are "neutralize[d]" [48]. So, even if an adversary has extensive auxiliary information about our data or has access to new information in the future, the Differential Privacy guarantee remains valid [50]. The section 5.3.3 provides an example of a successful linkage attack and how it could have been prevented if Differential Privacy was used.

This leads us to some of the properties of Differential Privacy that make it a popular technique to use in practice. One of the advantages of Differential Privacy is that it can be considered robust to post-processing. This means that without access to the private database, an attacker cannot derive further information about the participants by analyzing the output of the differentially private queries [48]. In addition, it is the only PET that takes into account the increased privacy risk that occurs when multiple queries are performed on the same database [20]. Thus, we can ensure that the Differential Privacy guarantee is upheld even when multiple analyses are performed [60]. Finally, we would like to point out that Differential Privacy also provides us with a metric to quantify the loss of privacy, allowing us to easily compare different algorithms and have more control over how much privacy we are willing to risk [55].

Legal & Regulatory Compliance As demonstrated in the last paragraph, Differential Privacy protects the specific information of individual participants. Therefore, it may be used to help companies and organizations to comply with applicable privacy laws and regulations while still being able to analyze and process private information [61]. However, it is not always clear whether the use of Differential Privacy mechanisms is even considered enough to comply with the necessary laws and regulations since the definition of privacy is subjective and regulations can be context-dependent [20]. We can refer to the paper "Differential Privacy: A Primer for a Non-Technical Audience" by Wood et al. [20] which discusses in more depth the challenges of Differential Privacy in the legal and regulatory context.

Versatility & Customizability Another advantage of Differential Privacy is its versatility and customizability. Part of the reason for this is that Differential Privacy is not a specific tool but a mathematical condition that can be satisfied by a wide variety of tools and algorithms [20].

This allows an analyst to select a tool ideally suited for their purposes. Furthermore, Differential Privacy is not limited to numeric data but can also be applied to non-numeric data, such as text, graph, image, or location data [62, 63]. As explained in the last section, Differential Privacy also has several parameters that can be adjusted as desired. There is not one optimal solution as to how to choose the parameters, as the requirements of different industries are vastly different [20]. For example, data used in recommendation systems do not require the same level of data protection as medical data. With Differential Privacy, we can change the epsilon parameter to determine how much privacy we are willing to risk [61]. It is even possible to adjust the delta parameter, which can provide a relaxation of Differential Privacy [48]. All of these factors combined result in Differential Privacy being versatile and customizable and thus applicable to a wide range of use cases.

Accessibility As with other PETs, the application of Differential Privacy leads to more companies and organizations being able to collect, process, analyze, and share information without violating laws and regulations. In addition, many people are less concerned about a

potential data leak because of the strong privacy guarantee. This could lead to greater consumer trust and more people being willing to share their data with third parties. Differential Privacy can also be used to publicly share data or data summaries in order to facilitate research and collaboration. For example, a hospital could publish medical data from a study, which could facilitate research by allowing more scientists to extract information from that data. [20]

Lightweight Computations & Data Preservation Finally, we would like to point out that Differential Privacy requires relatively little computing power compared to other PETs. This is because no complex computations have to be performed, but only random noise has to be added to the data. Moreover, Differential Privacy ensures that if Global Differential Privacy (GDP) is employed, i.e., only the query output is transformed, the original data is not lost. So, if the data is critical and must be preserved under all circumstances (e.g., medical records), you can use GDP to analyze the data without losing the original information. [64]

Challenges

The Optimal Trade-Off between Utility & Privacy One of the main challenges of Differential Privacy is the appropriate choice of the parameters and, thus, the struggle to find the optimal trade-off between utility and privacy. As discussed in the last section, choosing a low value for the epsilon-parameter results in higher privacy guarantees [64]. To achieve this higher level of privacy, more noise must be introduced into the data. However, this reduces the accuracy of the query outputs and the utility of the underlying data. Therefore, it is challenging to find a balance between fending off adversaries who want to do harm and providing analysts with useful information [64]. Finding an optimal solution for this trade-off is the focal point of current research [20].

Privacy Budget Allocation Setting a fixed Privacy Budget before analyzing sensitive data leads to another problem. In many cases, it is helpful to make several queries to a database in order to retrieve a great amount of information. However, as discussed in the last section, executing multiple queries successively reduces the Privacy Budget [20]. Therefore, in order not to compromise privacy, one must be content with a limited number of queries. However, this raises the question of which information should be prioritized [65]. In the case of the 2020 Census in the USA, it was necessary to assess which information is of particular importance to the general public and should hence be extracted from the data. However, these choices could be considered subjective, leading to unfair behavior towards certain groups of consumers using this published data [65].

At this point, we want to emphasize that this thesis is not focused on providing an overview of how to set the epsilon parameter or handle the Privacy Budget. However, we would like to point out that different methods, such as advanced composition [48], can help use the Privacy Budget more effectively.

High Level of Complexity Another problem is that Differential Privacy is a relatively complicated PET that can be difficult to explain. Since it is not a tool but a formal definition of a privacy goal, one must have various mathematical foundations to fully grasp the concept. Moreover, the notion of Differential Privacy is taught in only a handful of universities [65]. But even if one has a basic understanding of Differential Privacy, there are many other complicated problems to deal with. For example, choosing the right parameters for a specific use case or continuously monitoring the Privacy Budget across multiple queries are only some of the challenging tasks [20]. As a result, it is very hard to find enough experts in this field who have

the required expertise to build such a system from the ground up [65]. This can be a major obstacle to the widespread adoption of Differential Privacy.

However, lack of expertise is not the only problem. Currently, there are not many tools available to perform Differential Privacy or verify its correctness [65]. In addition, there is little to no guidance on how to deploy it due to a lack of standardization and best practices [50]. While tool support and standardization are issues that will hopefully be resolved in the near future, this is still a barrier for companies trying to adopt this technology today.

Another problem is that researchers are not used to dealing with data transformed by Differential Privacy. In the past, they had access to raw data, so they could analyze it for their own purposes without hesitation. With the introduction of Differential Privacy, they are now forced to adapt to this data by, for example, taking into account the margin of error that can arise from adding noise. [65]

In addition, the transformed data can look completely different than raw data. This can lead, for example, to even population figures being displayed as negative or fractional numbers. Since this is vastly different from what has been the norm in the past, this could lead not only to more confusion, but also to opposition from researchers who have benefited from unprocessed data in the past. For the 2020 Census in the USA, the Census Bureau wanted to fix this issue by converting negative numbers into non-negative ones using post-processing. However, it turned out that the post-processing step, intended to solve this problem, introduced further unnecessary errors in the data and even led to biases. This reduced the overall utility of the data. Therefore, the blog post "Implementing Differential Privacy: Seven Lessons From the 2020 United States Census" suggests keeping the transformed data as is and providing guidance to data consumers on how to handle it. [66]

As can be seen, Differential Privacy has several challenging aspects that have to be considered. Despite its great potential, its complexity is a hurdle to the widespread adoption of this technology.

Small Data Sets & No Individual Level of Analysis Another problem with Differential Privacy is that it is limited in its ability to handle small data sets. If you have a considerable sample size, the noise that needs to be injected is negligible. However, if you don't have enough data points, the amount of noise that would need to be introduced could heavily influence the results of the differential private analysis. Since this would greatly impair the utility of the data, you would hardly be able to elicit any meaningful information from it. [61]

Finally, it should be noted that in Differential Privacy, the analysis on an individual level is not possible. As we have noted several times, Differential Privacy prevents both an adversary and an analyst from finding out information specific to an individual. This makes it impossible to analyze the characteristics or identify a single participant. For example, Differential Privacy cannot help a bank identify malicious customers and their fraudulent transactions. [61]

5.3.3 Applications & Examples

The last sections were supposed to give the reader a basic understanding of Differential Privacy and its advantages and disadvantages. Now, however, we want to address where Differential Privacy can be applied in practice.

Areas of Application

Statistical Analyses First, one of the largest application areas of Differential Privacy is statistical analyses. A statistical analysis is the process of gathering and processing data to

obtain valuable insights and draw meaningful conclusions [67]. This can include the calculation of the mean, the median, or linear regression equations to extract valuable information from the data [20]. A notable example of a statistical analysis in which Differential Privacy was employed is the 2020 U.S. Census [66]. This census collected data from U.S. citizens to publish statistics such as the median household income and the poverty rate. Later in this section, this example will be discussed in more depth.

Machine Learning Differential Privacy can also be employed in machine learning to protect the privacy of the individuals that contributed their sensitive information to the training data. There are multiple ways of how to apply Differential Privacy in machine learning. For example, noise can be added during the training process to ensure that none of the sensitive information is being revealed. However, noise can also be added to the results of a query, thus slightly changing the decisions or prediction of the ML model. [50]

In the case of Federated Learning, Differential Privacy can be leveraged to transform the model updates that each client sends to a central server [68]. This ensures that even if an attacker gains access to the model updates, they cannot reconstruct the original data.

Ideal Use Cases

Although Differential Privacy has numerous applications in a wide range of industries, it is not a universal tool that should be used in every analysis. Therefore, in the following, we want to identify the characteristics of problems which indicate that Differential Privacy should be employed:

1. First, the data to be analyzed must be private, and there should be a need to protect it from being disclosed to unauthorized third parties. This should be obvious since the only goal of injecting noise into a data set is to ensure the privacy of the participants. Adding noise to non-sensitive data would only reduce its utility without providing any benefit. Furthermore, to ensure that the privacy of the participants is protected, the analysis isn't allowed to disclose specific data about individual data subjects. Therefore, it is only suitable for aggregate statistics and not for an analysis on an individual level. [20]

2. In addition, the data set being used should be large in size. As explained in the last section, if the sample size is large enough, the added noise is negligible, and one can still extract useful information from it. However, with small data sets, the query results can be heavily distorted [61]. In particular, with Local Differential Privacy, one relies on the law of large numbers to obtain a useful result [49]. Thus, if one has very few data points, one should not use Differential Privacy for analysis.

3. Finally, it should be noted that there should exist a reasonable trade-off between privacy and utility. Of course, anyone employing Differential Privacy seeks to maximize both privacy and utility. However, to increase utility, one would need to lower privacy expectations and vice versa. However, in some cases, both of them must be maximized. For instance, in the healthcare industry, privacy is paramount, but too much noise in the data could lead to poor accuracy. Since human lives are at stake when developing new drugs or treatments, having poor results is unacceptable. If neither privacy nor utility can be sacrificed, Differential Privacy may not be the best option for analyzing the data. [64]

Primary Industries

If a company has to collect and process a large amount of sensitive data, Differential Privacy can provide a strong privacy guarantee. Therefore there are a large number of industries that can benefit from the implementation of Differential Privacy.

For example, various technology companies usually gather and analyze a large amount of user data in order to gain more insights and improve their services. Organizations such as Google, Apple, Uber, and LinkedIn have already deployed it to ensure their users' privacy [69]. Another application area that can benefit from the introduction of Differential Privacy is the healthcare sector. As discussed in section 2.2, there are multiple regulations that prohibit the disclosure of medical records. Therefore, Differential Privacy could enable the processing of sensitive data and the collaboration between multiple parties.

Maturity & Adoption

Unlike Federated Learning, Differential Privacy has been around for quite some time. Since its inception in 2006, a great deal of research has been conducted, and thus, much theoretical knowledge has been gained. The current literature focuses on the development of new algorithms for different kinds of use cases and on the question of how to achieve the optimal trade-off between privacy and utility. [50]

Although there is a lot of theoretical knowledge on Differential Privacy, there are only a few implementations. Mostly large corporations like Google, Apple, and Uber have deployed it to ensure the privacy of their users [69]. However, it is not only used in the private sector but also utilized by government institutions that want to process or share data without violating the privacy of individuals [70].

Unfortunately, there is still a long way to go before Differential Privacy is widely adopted, as several challenges still remain, such as the high level of complexity and the current lack of standards. According to Gartner's "Hype Cycle for Privacy, 2021" [44], Differential Privacy will probably not achieve widespread adoption for another 5-10 years. However, an increasing number of tools are being developed to facilitate its implementation and deployment [20]. Similar to Federated Learning, organizations that choose to deploy Differential Privacy today may face several challenges but will also receive all of the advantages associated with early adoption.

Examples

A Preventable Privacy Breach Again, we will lead with an example of a privacy breach that could have been prevented if Differential Privacy had been employed.

In the mid-1990s, a state agency called Group Insurance Commission, which managed state workers' health insurance, wanted to release medical records in order to facilitate research [20]. They were aware of the possibility of a privacy violation if they simply published the collected data without redaction. Therefore, they first wanted to remove any identifiable information before publishing it. This included, for example, the removal of addresses or the replacement of names with random strings [16]. With these modifications, they assumed that the data would be secure, so they decided to make it available to the public. However, one thing they have not taken into consideration is the possibility of a linkage attack, as described in section 5.3.2. By comparing the data released by the Group Insurance Commission with the voter registration data, which was also publicly available, an MIT student was able to identify several individuals from the original data set, including the governor of Massachusetts [20, 16]. Obviously, this is a tremendous invasion of privacy since a lot of medical data, which can be

considered highly confidential information, was made available to the public. By applying Differential Privacy, the data would have been transformed in such a way that this type of linkage attack would not have been possible [16].

Successful Implementations Finally, we would like to demonstrate the potential of Differential Privacy by presenting two very well-known applications of this PET.

The U.S. Census Bureau operates as the largest government agency in the Federal Statistical System and is responsible for conducting the federal decennial census [65]. Its mission is to "serve as the nation's leading provider of quality data about its people and economy" [71]. The importance of this agency should not be underestimated, as census results are used to allocate the seats in the U.S. House of Representatives and help with the distribution of about $675 billion from the federal budget each year [66]. However, under Section 9 of the U.S. Code, the U.S. Census Bureau is also obligated to safeguard the participants' privacy [66]. Therefore, no data may be published that could help an adversary identify an individual in the data set. However, as a re-identification study for the 2010 Census revealed, existing privacy measures are not sufficient to provide this type of protection [70]. Therefore, Differential Privacy was employed in the 2020 Census to better protect the personal data of the participants. By deliberately injecting controlled noise into the data, they aimed to achieve high privacy guarantees [70]. However, implementing Differential Privacy on such a scale has been accompanied by various challenges. Discussing these is beyond the scope of this thesis. Instead, we can refer to Garfinkel et al. [65] and Hawes [66], in which they are discussed in more detail.

Apple, the largest corporation in the world, is another example of a company that wants to ensure the privacy of its users through the use of Differential Privacy. Apple is particularly known for the excellent user experience they offer to their customers. But they have to collect a large amount of user data to continuously improve their services. These services include, among others, typing and emoji suggestions and the identification of websites that drain the battery [72]. However, since most of the collected data can be considered sensitive, Apple has to guarantee its users that their privacy remains intact. For that reason, the company has chosen to deploy Local Differential Privacy [72]. This results in controlled noise being added to the data before it is even sent to a central server. This enables Apple to draw valuable conclusions about the whole user base without violating the privacy of the individual participants [72]. However, some external parties have also complained about their specific implementation of Differential Privacy, claiming that it does not sufficiently protect users' private information. For example, Tang et al. [73] identified several problems in Apple's implementation, such as resetting the Privacy Budget on a daily basis or generally setting the epsilon parameter too high. They demonstrate that Apple has made a step in the right direction but that there is still room for improvement.

5.4 Homomorphic Encryption

The SEBIS General Hospital has again encountered a privacy-related problem that we will first take another look at before we can start learning about the basics of Homomorphic Encryption:

SEBIS General Hospital has used Federated Learning and Differential Privacy to develop a novel ML model that can predict COVID-19 early in patients. However, after successfully developing it, they don't want to limit its application to their own patients but want to monetize this model by making it available to the general public. They, therefore, want to launch a website on which they offer Machine Learning as a Service (MLaaS). On this website, patients can upload their medical data and receive an accurate prediction based on this information for a small fee. However, this medical data is strictly confidential, and patients may fear that their data could get compromised. In addition, the hospital wants to avoid any potential liability and is therefore reluctant to collect this medical data. Consequently, it appears that SGH will not be able to realize this project. [74, 75]

However, this is not the case, as Homomorphic Encryption can be used to proceed with the project anyway. Homomorphic Encryption allows these patients to encrypt their data before it is transmitted to a central server [76]. Unlike traditional encryption schemes, this encrypted data can still be processed and used for the ML model. The resulting prediction of the ML model is also encrypted and cannot be decrypted until it reaches the patient [75]. In this section, we will discuss the definition and the properties of Homomorphic Encryption to provide a foundational understanding of this PET for a non-technical audience.

5.4.1 Definition & Characteristics

To start learning about Homomorphic Encryption, one must first have a fundamental understanding of general encryption and decryption. Nowadays, a large amount of sensitive data is sent over networks and stored in databases. Since this data has a high value, there are many malicious actors trying to gain access to this data illegally. To protect the data from these unauthorized third parties, it is possible to encrypt it. This involves converting plaintext, i.e., the raw data, into a format that unauthorized entities can no longer read [77]. This conversion is called encryption, and the resulting encrypted text is referred to as ciphertext [78]. In contrast, when the ciphertext is transformed back into plaintext to make it intelligible again, it is called decryption. In order to decrypt a message, one needs access to the corresponding decryption key. Depending on what type of keys are used, a distinction is made between Public and Private Key Encryption [77]. In the following, we will explain the difference between both of them by means of an example.

Alice wants to send Bob a private message, but she wants to ensure that no one else can read it, even if the message is intercepted during transmission. In Private Key Encryption, both encryption and decryption are done with the same key. This means that Alice first encrypts her message with the key c and that only Bob can decrypt the message with the same key c. However, this raises the question of how Alice and Bob can exchange their private key over an insecure channel. Fortunately, there are several key distribution protocols that make this possible. [77]

In contrast, Public Key Encryption has two keys, one public and one secret. The public key is openly accessible and can be seen by anyone. It can be used to encrypt a message. However, decryption only works with the secret key, which must be kept secret from unauthorized entities. In the case of Alice and Bob, Bob generates a public key and a secret key. The public key is accessible to everyone, including Alice. Alice can now use it to encrypt her message and send it securely over a network. But only Bob, with the secret key, which he has not shared

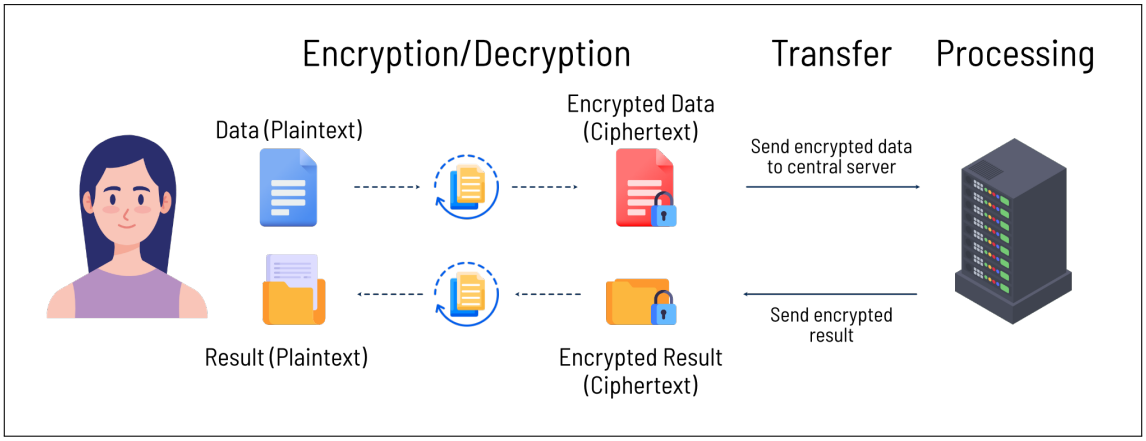


Figure 5.6: Homomorphic Encryption Process

with anyone, can use it to decrypt and read this message. Although this method does not cause problems regarding the secure exchange of keys, encryption and decryption require significantly more computing power. [77]

At this point, we do not want to go further into the details of general encryption and decryption. However, this short overview should suffice to start learning about the basics of Homomorphic Encryption.

Definition of Homomorphic Encryption

Now that we understand what encryption is in general, we can talk about Homomorphic Encryption. With traditional encryption methods, the encrypted data is protected when transmitted over a network or stored in a database [79]. However, if you want to process the data and gain valuable insights about it, you have to decrypt it first, which can lead to serious privacy issues. Homomorphic Encryption is an encryption scheme that can solve this problem, as the data remains encrypted even during processing [76]. This means that a third party not authorized to read the data can still perform useful operations on the ciphertext. These operations on the ciphertext reflect a corresponding operation on the plaintext [80]. As a result, after the ciphertext has been processed by an external party, it can still be sent to an authorized person, who then decrypts the ciphertext again and thus obtains the result of the operation [81]. This outcome is then identical to the result that would have been obtained if the operations had been performed on the plaintext instead. At no point was it necessary to disclose the raw data to the unauthorized entity actually processing it since the ciphertext does not reveal any information about the plaintext without the decryption key [82]. Figure 5.6 illustrates the concept of Homomorphic Encryption by means of an example.

Operations on Encrypted Data

Now the question arises of how performing operations on encrypted data can result in any useful outcome at all. In conventional encryption methods, the objective is to minimize the relationship between the plaintext and the ciphertext so that a malicious party cannot access any of the raw data. However, this also means that an operation on the ciphertext alters the plaintext in an unexpected way. In Homomorphic Encryption, on the other hand, the ciphertext should maintain the same structure as the plaintext so that the same mathematical operation produces an equivalent result. However, this is only possible if there is still a relationship between the plaintext and the encrypted data. The challenge is preserving the

structure as much as possible while preventing a malicious entity from gaining insight into the raw data by examining the ciphertext. Homomorphic Encryption can be considered successful if the decrypted result of adding two ciphertexts is identical to the result of adding the two corresponding plaintexts while simultaneously ensuring that the data cannot be read by an unauthorized party. [83, 84]

In the previous paragraphs, we have always referred to mathematical operations on encrypted data without explaining exactly what operations are allowed. In Homomorphic Encryption, generally, only addition and multiplication are possible [82]. At first glance, this seems very restrictive. However, addition and multiplication are sufficient to represent or approximate any mathematical function [21]. This allows complex operations to be performed. However, complex operations can be more difficult to execute than usual, as the calculation needs to be broken down into addition and multiplication operations only [82].

If an addition is homomorphic, then the result from adding two ciphertexts is equal to the encrypted result from adding the two corresponding plaintexts [83]. Since the same is true for homomorphic multiplication, the homomorphic property of these operations can be described mathematically as follows [21]:

$$\begin{aligned} \text{Enc}(a) + \text{Enc}(b) &= \text{Enc}(a + b) \\ \text{Enc}(a) \cdot \text{Enc}(b) &= \text{Enc}(a \cdot b) \end{aligned}$$

Here $Enc(x)$ represents the encrypted value of the plaintext x. The following shows that the results of the operation on the encrypted data can be decrypted. The resulting output is identical to the result you would have gotten if you had just performed the operations on the two corresponding plaintexts [21]:

$$\begin{aligned} \text{Dec}(\text{Enc}(a) + \text{Enc}(b)) &= \text{Dec}(\text{Enc}(a + b)) = a + b \\ \text{Dec}(\text{Enc}(a) \cdot \text{Enc}(b)) &= \text{Dec}(\text{Enc}(a \cdot b)) = a \cdot b \end{aligned}$$

This shows that if an encryption scheme is homomorphic, the operations performed on the plaintext can also be performed on the encrypted data [83]. Thus, one can outsource computations to an unauthorized third party without granting access to the raw data.

Even though we have only talked about addition and multiplication, this does not mean that non-numerical data types cannot be used for Homomorphic Encryption. For example, it is possible to first convert plain text into numbers using ASCII, on which you can then perform Homomorphic Encryption. For instance, one can employ Homomorphic Encryption to determine whether a word is contained within a paragraph [85]. Neither the search word nor the paragraph has to be disclosed to the entity performing the search.

Noise in Homomorphic Encryption

At this point, we would like to mention that the transformation from plaintext to ciphertext involves adding noise to the data, making it impossible for malicious entities to read it. The task of decryption is to then remove this noise from the ciphertext so that the one with the corresponding key can access the plaintext again. However, operations on the encrypted data increase the noise of the ciphertext, thus making it more and more difficult to decrypt it successfully. In this context, multiplication increases the noise much more than addition. If the accumulated noise is too large, the ciphertext may no longer be able to be decrypted. In this case, only a limited number of operations should be performed on the data. [81, 86]

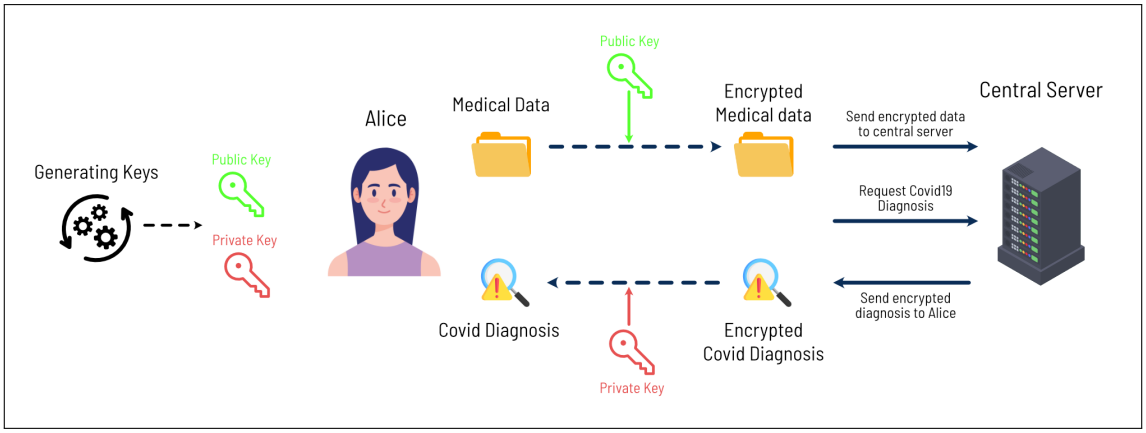


Figure 5.7: SGH's Homomorphic Encryption Scheme (Based on [21], [87])

There are several different types of Homomorphic Encryption. They differ in how often and what operations can be performed on the ciphertexts. For example, in Fully Homomorphic Encryption (FHE), both addition and multiplication can be performed infinitely often [87]. Later in this section, these different types of Homomorphic Encryption will be presented in more detail.

Homomorphic Encryption Process

Although we have already discussed how the encryption process works in general, we would now like to outline an entire Homomorphic Encryption process by means of an example. This example is illustrated in Figure 5.7 and based on the Homomorphic Encryption process described by Acar et al. [21] and Awadallah et al. [87].

Because Alice is afraid that she may have contracted the Coronavirus, she wants to use the SEBIS General Hospital's (SGH) MLaaS product that can predict COVID-19 early. She is confident that her data is protected due to the Homomorphic Encryption scheme employed by the SGH. Since Alice wants to use Public Key Encryption, she starts by generating a public key and a secret key. Using the public key, she can then encrypt her medical data appropriately before sending it to a central server for storage. After she requests the server to apply the ML model, a prediction is made by performing operations on the encrypted data. With conventional encryption schemes, this last step would not have been possible. The server would require the corresponding secret key to decrypt the data before it could perform any useful operations on it. However, in Homomorphic Encryption, the decryption of the data for processing is not required. Finally, Alice receives the encrypted result from the server and can decrypt it by using her secret key to obtain the prediction. In this example, Homomorphic Encryption was employed so that the plaintext is never disclosed to a third party. The hospital was able to make a prediction by using Alice's medical information without ever having access to her raw data.

It should be noted that although Homomorphic Encryption protects the confidentiality of the data, it does not guarantee its integrity [88]. This means that the data cannot be accessed by others, but the person processing the encrypted data can deliberately give you an incorrect answer to your request. However, there are several ways to ensure integrity, for example, with the help of Verifiable Computation [88].

Types of Homomorphic Encryption

As mentioned earlier, there are several different types of Homomorphic Encryption schemes. They are categorized according to their capabilities and computational complexity [87]. In the following, we briefly outline the different types of Homomorphic Encryption.

Partial Homomorphic Encryption: In Partial Homomorphic Encryption (PHE), either addition or multiplication can be performed infinitely often on the ciphertext, but not both [89, 84]. Designing a PHE scheme is relatively easy, such that even some already well-known encryption schemes just happened to be PHE [84].

Somewhat Homomorphic Encryption: In contrast, Somewhat Homomorphic Encryption (SWHE) allows for both addition and multiplication, but only for a limited number of operations [87]. If the number of operations exceeds a certain limit, the accumulated noise becomes too high, and the ciphertext cannot be decrypted again [87, 81]. It is therefore mainly used for elementary functions or simple statistical analyses of encrypted data [89]. The design of a SWHE scheme is already more challenging than that of a PHE scheme. [84]

Fully Homomorphic Encryption: In Fully Homomorphic Encryption, both addition and multiplication can be performed on encrypted data an infinite number of times [87]. This means that any kind of computation can be performed on the encrypted data [89]. Although FHE is the most powerful Homomorphic Encryption scheme, it can't be used all the time since it is very computationally intensive. This makes it unfeasible for many real-world use cases.

The first successful Fully Homomorphic Encryption algorithm was developed by Craig Gentry in 2009 [90]. However, since performing operations on the ciphertext was much slower than on the plaintext, many other algorithms had to be developed to improve efficiency. But the huge computational overhead is still a challenge that one has to face when using a Fully Homomorphic Encryption scheme [84].

All of these different types of Homomorphic Encryption have different capabilities and limitations and, thus, their own particular use cases. However, we will not elaborate further on this categorization here. We can refer to Awadallah et al. [87], where this is discussed in more detail.

5.4.2 Benefits & Challenges

Since we have discussed the features of Homomorphic Encryption in the previous section, we would now like to explore its capabilities and limitations in more depth. First, we will look at the most notable advantages of Homomorphic Encryption before moving on to the various challenges.

Benefits

Privacy The major advantage of Homomorphic Encryption is that it guarantees the privacy of sensitive data - even if it is being processed by an untrusted third party. With traditional encryption schemes, data can only be protected while at rest in a database or in transit [79]. However, if one were to perform operations on this ciphertext, one would not get any sensible results [21]. Hence, in this case, the data has to be decrypted first before one can process it in a meaningful way. This is often an obstacle for many organizations, as they fear their data might get compromised [91]. In contrast, Homomorphic Encryption allows you to protect this

data from unauthorized parties throughout the entire data lifecycle [83]. It allows data to be processed even if it is encrypted, so there is no need to ever disclose one's data to the entity responsible for processing it. Only people with access to the secret key can then decrypt the resulting ciphertext [83].

Legal & Regulatory Compliance Third parties who store and process encrypted data can also comply more easily with the various data protection laws and regulations [83]. Since they themselves do not have access to their customer's sensitive data, they cannot leak it either and thus do not expose themselves to any liability [74]. In the SEBIS General Hospital example, their goal is to monetize their MLaaS offering, not to collect their customers' sensitive data. That's because if they were to store their customers' raw medical data, they would be responsible for protecting it to avoid non-compliance with regulations such as HIPAA. It should be noted that Homomorphic Encryption can help an organization comply with only a subset of these regulations.

Quantum-Safe Security Although one's private data may be secure today, it does not mean it will be in the future. In the wake of quantum computing, many enterprises are concerned about the safety of their current encryption schemes. But fortunately, Homomorphic Encryption can be considered quantum-safe and is thereby highly secure against any attempts to break it. [92]

Secure Outsourcing of Computation Another advantage is that the homomorphic property of this encryption method allows companies that previously had to forgo the outsourcing of computation due to privacy concerns to finally take advantage of it [83]. The main reason why many organizations are reluctant to embrace cloud computing is the fear of potential security concerns [87]. Since cloud service providers can store and process data for you, it is also possible for them to access that data [80]. For instance, some of the cloud provider's employees may gain access to the data. Even one of the largest cloud providers, Google, has been forced to terminate employees for accessing customer data without proper authorization [80]. Incidents like this discourage organizations from utilizing these types of services. However, Homomorphic Encryption can now be applied to alleviate the fear companies have regarding security concerns. Because by using this encryption method, organizations can outsource their computation without the raw data being passed on to a third party. An unauthorized entity can perform operations on the encrypted data and return the resulting ciphertext that can only be decrypted by the data owner [83]. Thus, even highly sensitive data can be processed by an external party without being compromised. Homomorphic Encryption is therefore an ideal solution for companies to gain the trust of their customers and can even be leveraged to enable new business models [93] (see the example of the SEBIS General Hospital in section 5.4).

Challenges

Even though Homomorphic Encryption has many advantages, there are currently also many hurdles that must be overcome before it can be employed successfully.

Performance Overhead The most significant disadvantage of Homomorphic Encryption is its inefficiency, which leads to a huge performance overhead [91, 80]. As a result, Homomorphic Encryption is not suitable for many real-world applications.

How inefficient it is becomes clear when comparing a computation performed on the plaintext with the same computation performed on the ciphertext. The first version of the

Fully Homomorphic Encryption scheme developed by Craig Gentry performed operations on the ciphertext slower than operations on the plaintext data by a factor of about a trillion [80].

Since then, amazing progress has been made to improve the efficiency of this encryption scheme. But even today, the performance overhead is still considered a significant barrier to adoption, and even IBM states that operations can still be hundreds of times slower when performed on encrypted data [79].

However, it should be noted that not all types of Homomorphic Encryption schemes have the same computational complexity. The computational overhead will vary significantly depending on which Homomorphic Encryption method you choose. For instance, Partial Homomorphic Encryption is considered to be an efficient encryption algorithm and has already been widely used in practice [87]. However, since PHE only allows either multiplication or addition operations, the number of possible use cases is limited [87]. If you want to be able to perform any computation, you must implement a Fully Homomorphic Encryption scheme. However, as mentioned earlier, FHE leads to a huge performance overhead, which makes it unfeasible for many real-world use cases [80]. Nowadays, much research is devoted to making Homomorphic Encryption more efficient and thus more useful for many real-world applications [84].

High Level of Complexity Another drawback of Homomorphic Encryption is that it can be considered a complex technology. Developers without prior knowledge of cryptography might have a hard time grasping the concept of Homomorphic Encryption. In addition, programming a function to operate on raw data is not the same as doing so for encrypted data. Developers may not be familiar with best practices when it comes to programming on ciphertexts. All of these factors combined lead to complexity being a hindrance to the widespread adoption of Homomorphic Encryption. [79, 91]

Scalability Barriers In addition, Homomorphic Encryption does not scale well, making it impractical for large data sets [91]. This is due not only to the performance overhead mentioned earlier but also to the different sizes of plaintext and ciphertext. In fact, according to Mark D. Ryan [80] the ciphertext requires significantly more storage than the corresponding plaintext. Therefore, it is not feasible to encrypt large amounts of data or even an entire database with Homomorphic Encryption [91]. Together with the increasing computational overhead, this results in Homomorphic Encryption not being scalable. Therefore, it should only be used for relatively lightweight, privacy-sensitive data that should not be exposed to the third party handling it [91].

Constraints on Operations Another disadvantage of Homomorphic Encryption is the limited operations that can be performed on encrypted data. As explained in the last section, different types of Homomorphic Encryption schemes have varying capabilities. For instance, Fully Homomorphic Encryption allows for arbitrary computations on encrypted data, but at the cost of performance [89]. Since this trade-off is unacceptable for many use cases, one has to resort to other Homomorphic Encryption schemes. But most of them have severe limitations regarding the operations that can be performed on encrypted data, limiting their practical utility [21, 91]. In summary, Homomorphic Encryption is too restrictive for many real-world applications, either because of the limited operations or because of the increased storage and performance overhead.

5.4.3 Applications & Examples

Up to this point, the goal has been to provide a fundamental understanding of the characteristics, possibilities, and limitations of Homomorphic Encryption. In this section, we shift the focus to its most significant applications.

Areas of Application

Secure Cloud Computation One of the largest application areas is Secure Cloud Computation [94]. As discussed in the last section, the drawback of traditional encryption schemes is that the data has to be decrypted before operations can be performed on it [89]. This discourages many companies from leveraging cloud computing. With the advances in Homomorphic Encryption, however, sensitive data can now be processed by a third party, even when encrypted [94]. This allows companies to leverage the power of the cloud without compromising privacy. For instance, the healthcare and financial sectors can use it to analyze sensitive data in order to derive valuable insights, even if they do not have sufficient computing infrastructure themselves.

Secure Machine Learning Secure Machine Learning is yet another crucial application of Homomorphic Encryption. ML models often have to be trained on sensitive data. In this case, one would like to prevent this data from being shared with the model developers. To achieve this, Homomorphic Encryption can be used to train the ML model on encrypted data, ensuring that the raw data is never disclosed to the third party developing it [94]. Especially in the healthcare and finance sector, where privacy is a top priority, this allows predictive models (e.g., for disease and fraud detection) to be built by external parties without compromising privacy [94]. Furthermore, ML models can be used to make predictions based on encrypted data, even if they have been trained on plaintext [75]. The resulting prediction is also encrypted and can only be decrypted by the person holding the secret key. We have already provided an example of this in section 5.4 about the SEBIS General Hospital. For more information, see Dowlin et al. [75], where this application of Homomorphic Encryption is explained in more detail.

Supply Chain Security Another notable application of Homomorphic Encryption is Supply Chain Security. Large enterprises, in particular, rely on multiple third parties, such as contractors or suppliers, to keep their supply chain and, thus, their business running smoothly [89]. Sharing data with these third parties raises new security concerns for the organizations [84]. Because regardless of how many security measures a company has in place, sharing raw data with a vendor that has lower security standards increases the risk of a possible privacy breach. To mitigate this increased risk, Homomorphic Encryption can be employed to ensure that only encrypted data is sent to a third party [84]. So even if there is a privacy breach at an external vendor, the attackers cannot access any of the organization's raw data since they have only shared encrypted data. Thus, Homomorphic Encryption can alleviate potential privacy risks and increase the level of security in the supply chain [94].

Ideal Use Cases

Similar to other PETs, Homomorphic Encryption is not applicable in all scenarios and does not necessarily represent the optimal encryption scheme for every use case. We, therefore, want to identify the characteristics of problems that strongly suggest that Homomorphic Encryption is the right PET to solve them:

1. First of all, Homomorphic Encryption should only be used on data that is sensitive and that should not be disclosed to an unauthorized party under any circumstances. If this is not the case, Homomorphic Encryption would only introduce a significant storage and performance overhead without any good reason [80]. Moreover, it is also not advisable to employ Homomorphic Encryption if the third party collecting and processing the data can be fully trusted and there are no other ethical or legal considerations [95]. In this case, many more efficient encryption schemes can protect the data in transit and at rest.

2. Furthermore, Homomorphic Encryption should not be applied to data that is large in size, nor can too overly complex operations be performed on the encrypted data. As discussed several times, Homomorphic Encryption substantially increases storage requirements since the ciphertexts tend to be significantly larger than the corresponding plaintexts. Thus, encrypting a large amount of data or even an entire database is not feasible. Furthermore, it is safe to say that operations that already take a long time to perform on plaintext are not suitable for Homomorphic Encryption. In summary, with Homomorphic Encryption, one should focus on the most critical data and refrain from performing too overly complicated operations. [91]

3. In Homomorphic Encryption, the results of the operations performed on the encrypted data remain encrypted and cannot be decrypted without the corresponding secret key. Although this may initially appear to be only advantageous, there are also some use cases where this is a hindrance. That's because the third party processing the encrypted data is not able to make any decisions based on the encrypted results. So, in this case, the result must first be returned to the data owner, who can then decrypt them and take new actions based on them. However, this process requires a lot of time and computing power, which has to be avoided in several application scenarios. For example, one would like to filter out spam mail automatically with the help of a cloud service. Since one does not want to disclose the content of the mail to the cloud provider, one decides to employ Homomorphic Encryption. However, the cloud service cannot see the result of its own prediction, as it is also encrypted. This means that it cannot automatically decide whether the email should be moved to the spam folder. The encrypted result has to be first sent back to the data owner in order to be decrypted. In summary, the cloud service can detect spam but cannot discard spam automatically. If one wants the third party processing the data to make decisions based on the results, Homomorphic Encryption is not the technology to choose. [80]

Primary Industries

Homomorphic Encryption can be useful in various industries as it ensures that one can outsource computations to third parties without disclosing any raw data. This PET is particularly advantageous in sectors with highly sensitive data subject to numerous laws and regulations governing how it should be dealt with [96]. Therefore, healthcare is one of the sectors that can greatly benefit from the introduction of Homomorphic Encryption. The reason for this is that there is a lot of sensitive patient information that should not be exposed under any circumstances but that is needed to make further advances in the healthcare industry [86]. For instance, it could be used to identify trends in medical records or to develop ML models that can help detect or even predict common illnesses [79, 86]. The finance sector is another notable example of an industry that can use Homomorphic Encryption to improve its services. For example, an ML model could be trained to detect malicious transactions without the need to use customers' sensitive data [79].

Maturity & Adoption

The concept of Homomorphic Encryption has already existed for more than 34 years. Even then, performing operations on encrypted data was already seen as extremely useful. However, the first Fully Homomorphic Encryption scheme was not developed until 2009 by a graduate student, Craig Gentry. Even though it worked in theory, his FHE scheme could not be applied in the real world due to the enormous computational overhead it introduced. [21]

Since then, much research has been done to improve the efficiency of Homomorphic Encryption. Companies like IBM and Microsoft have already built libraries for Homomorphic Encryption and are working on their continuous improvement [83]. For instance, IBM released a new version of the open-source Homomorphic Encryption library HElib in 2018 that was already about 2 million times faster than the original version [83]. Despite already huge improvements, Homomorphic Encryption is still inefficient for many use cases. Hence, Gartner estimates that Homomorphic Encryption will still need a time-to-market of about 3-6 years [97].

When deploying HE today, one must be aware of the various challenges. However, it can already be used in practice, especially for efficient operations on sensitive data that should not be disclosed to third parties [91].

Examples

A Preventable Privacy Breach First, we would like to consider an example of a privacy breach that could have been prevented through the deployment of Homomorphic Encryption.

The Cambridge Analytica scandal revealed that the data of 50 million Facebook users was harvested, with many of them never granting access to their own personal data. This was made possible through the *thisisyourdigitallife* app, which leveraged people's Facebook information (e.g., name, emails, interests, preferences, etc.) to create a personality profile for them. However, not only the data of the app's users but also that of their Facebook friends were collected without them being informed about it. All of this data was stored and later sold to the data analytics company Cambridge Analytica. This company then utilized this user data to create behavioral profiles for targeted advertisement in the 2016 Presidential Election and Brexit. This could have been prevented if Facebook had mandated Homomorphic Encryption. That way, users of the app *thisisyourdigitallife* would only need to provide encrypted data. Due to the properties of Homomorphic Encryption, operations could still have been performed on the encrypted data to build the desired personality profile. However, the raw personal data would not have been disclosed to an unauthorized third party. [98, 16]

Successful Implementations There are only a limited number of successful applications of Homomorphic Encryption in the real world. Therefore, we would like to take a closer look at a pilot project of the bank Banco Bradesco S.A. instead and then explore the potential of Homomorphic Encryption for Facebook.

One of Brazil's largest banks, Banco Bradesco S.A., has joined forces with IBM to run a pilot project [99, 100] to implement Homomorphic Encryption. Their goal was to apply this encryption scheme for machine learning tasks. In the financial sector, machine learning can be used to build various predictive models, e.g., for fraud detection or risk assessment. The bank already had a machine learning model that allowed them to predict when customers would need a loan in the near future. To make this possible, bank employees usually have to manually select the variables (e.g., the amount of money they spend on groceries) that best indicate this. When selecting these variables, they gain access to the customers' raw data, which could

lead to privacy concerns. But with IBM's help, this process can be performed more securely by applying Homomorphic Encryption. Thus, no private information had to be revealed in order to identify the variables that should be included in the prediction system. In addition, they employed Homomorphic Encryption to show that they could make predictions based on encrypted data and thereby obtain an encrypted result. This allows them to outsource this prediction task to an external service provider without compromising the confidentiality of the raw data. However, we would like to reiterate that this was not a change to all their systems but only a pilot project to explore the potential of Homomorphic Encryption. [100]

Facebook is one of the largest US technology companies and has reportedly already invested in the research of Homomorphic Encryption [101]. At the time of writing this thesis, Facebook is not currently deploying Homomorphic Encryption for its applications, but it is obvious that it has a lot of potential for the future. For example, they could use it for targeted advertisements by first encrypting users' data before making decisions about the most suitable ads based on it [102, 101]. Thus, sensitive data (e.g., the last websites visited) would not have to be sent to a central server of an advertising company. Another notable example is the use of Homomorphic Encryption for Facebook's Messenger apps, such as Whatsapp, which would allow messages to be analyzed for harmful content or spam detection without revealing sensitive information about users [80, 103]. As you can see, there are many applications that Facebook can consider in the future. However, it might still take years before they manage to successfully integrate this PET into their applications [101].

5.5 Secure Multi-Party Computation

First of all, we would like to take another look at a problem encountered by the SEBIS General Hospital before introducing the fundamentals of Secure Multi-Party Computation (SMPC):

The SEBIS General Hospital wants to be able to identify patients with a higher risk of getting breast cancer through genetic testing and a genome-wide association study (GWAS). However, to identify potential correlations between genetic variants and a specific disease, one needs a large amount of data to make a conclusive statement. Since the SGH is still in need of enough medical data, it agrees to enter into a partnership with the Logan General Hospital, which is also convinced of the value of a large-scale GWAS. However, genomes are considered highly sensitive, as they can be used to obtain a large amount of personal information that should not be disclosed under any circumstances. Due to the various regulations governing the protection of this kind of medical data, it wouldn't be possible for the two hospitals to pool their raw data and then analyze it. Therefore, they would not be able to conduct this crucial research in order to protect the privacy of their patients. [104, 105, 106]

However, they can use Secure Multi-Party Computation to perform this life-saving research nonetheless. SMPC allows the hospitals to use their sensitive data to perform a joint computation without the need to reveal any of their raw data [107]. In the following section, we would like to give an overview of the basic features and characteristics of Secure Multi-Party Computation to provide a fundamental understanding of how it can be leveraged to ensure privacy. The approach used to introduce this PET is derived from Lindell [22].

5.5.1 Definition & Characteristics

There are numerous situations where one would like to cooperate to achieve a common goal but is not comfortable disclosing one's data. As shown in the example of the SGH, this can result in analyses not being carried out due to privacy concerns. For that reason, Secure Multi-Party Computation was created to enable multiple parties to perform a joint computation without the need to reveal any of their private inputs in the process [107].

The advantage of SMPC is that you don't have to entrust your data to a central entity [108]. The calculation is carried out by the participants without them having to put their privacy at risk. Undoubtedly, this PET offers many opportunities in industries in which secure collaboration is paramount. To better understand what this entails, we would first like to start with a simple example.

Example: Sharing Salary (Based on: [109])

Alice ($80,000), Bob ($60,000), and Charlie ($70,000) want to calculate their average salary. Due to privacy concerns, however, they do not want to disclose how much they earn. Therefore, each person divides their salary into three randomly chosen shares, the sum of which corresponds again to the actual value of the salary. For instance, as shown in Figure 5.8, Alice's salary is divided into the following three shares:

Share 1: $\$20,000$

Share 2: $-\$100,000$

Share 3: $\$160,000$

Share 1 + Share 2 + Share 3 = $\$20,000 + (-\$100,000) + \$160,000 = \$80,000$

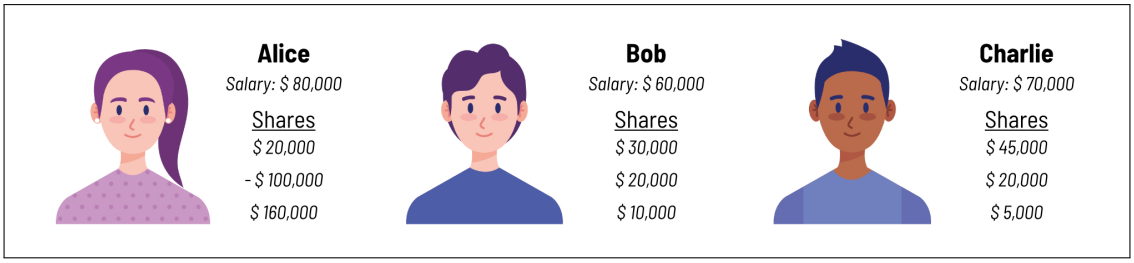


Figure 5.8: Salary Sharing - Dividing Salaries into Random Shares (Based on [109])

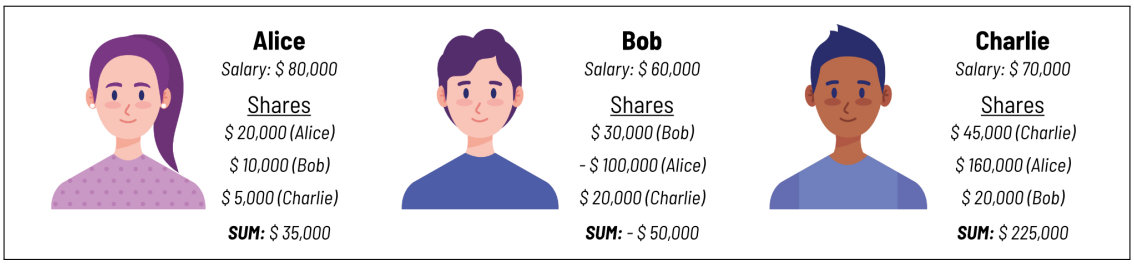


Figure 5.9: Salary Sharing - Distributing Shares Among Participants (Based on [109])

As illustrated in Figure 5.8, Bob and Charlie do the same with their salaries.

Then Alice, Bob, and Charlie give one of their shares to each of the other participants so that, in the end, everyone has a total of three shares (see Figure 5.9). Although everyone now has a share of each participant's salary, none of them can determine the true value of the others' salary. However, they are now able to calculate their average income.

Alice, Bob, and Charlie can each sum up the shares they have right now and can then share this intermediate result with the other participants. They can then calculate the average as you would without SMPC:

$$\frac{\text{Result Alice} + \text{Result Bob} + \text{Result Charlie}}{3} = \frac{\$35,000 - \$50,000 + \$225,000}{3}$$
$$= \frac{\$210,000}{3} = \$70,000$$

Requirements of an SMPC System

The previous example demonstrated how SMPC can be leveraged to calculate a function without the participants revealing their private inputs. However, not all entities participating in these kinds of processes can be trusted. Malicious third parties may try to falsify the result of the calculation or even determine the private inputs of the participants [110]. SMPC protocols, which define the exact procedure of the SMPC process and the desired behavior of the participants, must therefore take into account the dishonest behavior of a potential opponent [108]. These adversaries have various capabilities to attack SMPC tasks and are even able to control a number of the participants for their purposes [108]. These participants are then referred to as corrupted and obey the instructions of the adversary controlling them [22].

To enable a secure collaboration despite these potential attacks, protocols trying to achieve SMPC should at least meet the following criteria:

Privacy First, it must be ensured that each participant only has knowledge about their own input, the output of the entire SMPC process, and the information that can be inferred from both of them [108].

Correctness Secondly, the SMPC process should never yield a falsified result. This means that an adversary should not be able to manipulate the protocol execution in such a way that it produces a false output [110].

Independence of Inputs Next, the inputs of the corrupted participants must be selected independently from the honest participants' inputs [22].

Guaranteed Output Delivery Furthermore, an SMPC protocol must prevent corrupt entities from denying honest participants access to their own results [22].

Fairness Finally, the SMPC protocol should be fair, i.e., the corrupted parties only obtain their outputs if the honest participants also obtain theirs [108].

Real-Ideal Paradigm

So far, we have repeatedly discussed the importance of security in the context of SMPC without explaining its definition. In the previous paragraph, we listed several criteria associated with the term security. However, a list of properties that should be fulfilled is not a sufficient definition of a secure protocol, as it is not only an arduous task to create that list but also extremely easy to overlook certain security aspects entirely. Therefore, the real-ideal paradigm was developed to offer a relatively straightforward definition of security in connection with SMPC protocols. [22, 107]

In the real-ideal paradigm, we first assume an ideal world in which we want to compute a function together with several other parties. In this ideal world, there is also an external party that all participants can trust completely. Therefore, each participant can just pass on their private data to this external party, which then performs the requested computation. When finished, the trusted party distributes the results to all participants. Since this computation is not performed by all the participants but by an external party, the only thing a potential attacker could do is manipulate the inputs of the corrupted participants. [107, 108]

Even if such an ideal world would be perfect for SMPC, this is unfortunately not achievable, as there is no fully trustworthy party in the real world. Therefore, computations must be performed between all participants, which leaves each of them with more responsibility. This results in more opportunities for the adversary to attack the protocol execution by controlling a number of the participants. [22]

The question arises as to why we even discuss an ideal world if it is vastly different from reality and cannot be implemented. The notion of an ideal world can help us define the concept of security by providing a reference point against which we can compare the security of SMPC protocols. This means that a protocol can only be considered secure if a potential attacker has the same opportunities to negatively affect the SMPC process in the real world as would be possible in the ideal world. An SMPC protocol should therefore offer the exact same security guarantees as is provided in an ideal world. [107, 108]

It can be shown that the SMPC process in an ideal world satisfies all the requirements listed above. If interested, we can refer to Lindell [22], where this is discussed in more depth.

Threat Models

In the previous paragraphs, we always acknowledged the existence of adversaries with varying capabilities to attack the SMPC process. Now we would like to elaborate on this issue by describing the different types of adversaries:

Semi-honest adversaries: We can refer to an adversary as semi-honest or honest-but-curious if they try to obtain private information from the other participants but do not deviate from the specified protocol. This includes gathering information (e.g., messages sent by honest parties) from a number of corrupted participants during the execution of the SMPC process. In that case, the adversary doesn't actively interfere with the protocol execution and is often also known as a passive attacker. [107]

Malicious adversaries: In contrast, a malicious attacker can disregard the given protocol in its entirety and instruct the corrupted participants to diverge from the defined SMPC process without restriction. If a protocol is secure against such adversaries, then it can withstand any potential attacks that could happen during the SMPC process. However, these protocols tend to be a lot less efficient, which renders them unsuitable for some real-world applications. [108]

Covert adversaries: If one is dealing with malicious attackers but not willing to put up with the computational overhead introduced by a fully secure SMPC protocol, then one can think of these attackers as being covert adversaries. Although these attackers are able to deviate from the protocol, there is a certain probability that they will be detected as malicious entities. Since malevolent attackers can face serious consequences in real life, the adversaries must consider whether it is even worth trying to attack the SMPC process. [108]

Having established what types of attackers exist, we would now like to discuss the various corruption strategies they can employ to target the SMPC process, as described by Lindell [22]. First, one could assume that the corrupt participants are already fully determined before the start of the protocol execution and that they also remain corrupt during the course of the computation. Participants who were honest at the beginning will also be honest throughout the entire SMPC process. This is known as the static corruption model. In contrast, in the adaptive corruption model, the adversary can gain control over the participants even during the execution of the protocol. Thus, honest parties can be corrupted over time. Finally, there is the proactive security model, where corrupt parties can become honest during protocol execution, and honest participants can get corrupted. Again, we can refer to Lindell [22], who discusses these corruption models in more depth.

Depending on which malicious party is being dealt with, a different implementation of SMPC may be required. However, more stringent security requirements often correlate with higher computational complexity [107]. Therefore, a trade-off between security and efficiency is often unavoidable.

Basic Cryptographic Primitives

A cryptographic primitive can be seen as a "basic building block" [111] for more complex cryptographic schemes or protocols. In the following, we would like to briefly elaborate on the three most common ones used in SMPC protocols. However, we are aware that the concept behind these primitives can be quite complex and not suitable for a non-technical audience. Therefore, they will not be discussed further in this chapter after the following paragraphs.

Shamir's Secret Sharing (SSS) is the concept of dividing a secret into shares and then distributing them among n entities. However, each share of the secret does not reveal any useful information about the secret by itself. In a SSS scheme, often denoted as (t, n)-SSS, at least t participants must work together by combining each of their shares to reconstruct the secret. For instance, in a (3,3)-SSS scheme, there are three participants, each of whom receives

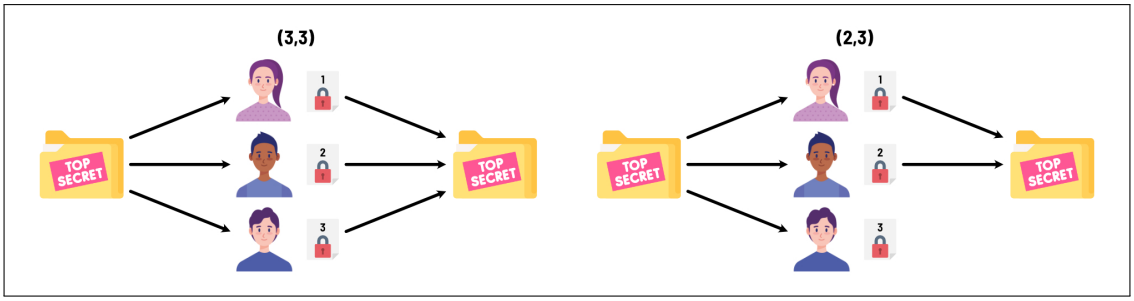


Figure 5.10: Shamir's Secret Sharing

a share of the secret s. All three participants are then needed in order to combine their shares to reveal s. Hence, either all participants learn about the secret, or no one does. In contrast, in a (2,3)-SSS scheme, although there are three participants, only two of them must combine their shares in order to reconstruct the secret. For an illustration of this cryptographic primitive, see Figure 5.10. [107, 112]

Oblivious Transfer (OT) is another primitive that can be used as a basic building block of an SMPC protocol. The most commonly used definition involves two parties and can be referred to as 1-out-of-2 OT. The aim of Oblivious Transfer is for an interested party (B) to learn more about parts of another participant's (A) secret information. In this context, A has two secrets, s1 and s2, that could be sent to the interested party. However, B now has the choice of receiving either s1 or s2, but not both. Additionally, party B does not want A to gain any information about this procedure (e.g., which message B selected). [107]

Homomorphic Encryption (HE), the Privacy-Enhancing Technology discussed in the last section, is another cryptographic primitive that can be used for SMPC protocols. However, in this context, a special kind of HE scheme must be applied to allow for multiple participants. For instance, Multi-Key Homomorphic Encryption can be employed to enable multiple parties to compute a joint function. Each participant uses their own public key to encrypt their sensitive data and then sends it to a central server. After computing the requested function based on the participants' encrypted input, the server sends the clients the encrypted result of the computation that can be decrypted by each involved party collaboratively using their own private key. [108]

It should be noted that these are highly simplified explanations of these primitives. An in-depth discussion of them can become very complex and is, therefore, beyond the scope of this thesis. If interested, we can refer to Evans et al. [107], where this is discussed in greater length.

SMPC Protocols

Although a detailed description of the specific SMPC protocols would go far beyond the scope of this work, at least for the sake of completeness, we would like to mention the most relevant one.

Yao's Garbled Circuits, was introduced by Andrew Yao [113] in 1986. It is a Secure Two-Party Computation protocol, i.e., it allows two parties to perform a joint computation without disclosing their private inputs [114]. It is based on Oblivious Transfer and secure against honest-but-curious adversaries [108]. It is well-known for its ability to solve the millionaire problem, in which two millionaires want to figure out who has more money without revealing their net worth [115]. Even though the protocol is more than 30 years old, it is still highly regarded for its efficiency [107]. Several protocols that were developed later on are still based

on Yao's Garbled Circuits.

Other essential protocols include the Goldreich-Micali-Wigderson Protocol and the Ben-Or, Goldwasser, and Wigderson Protocol. Since a further discussion of these protocols is outside the scope of this thesis, we would like to refer to Evans et al. [107], where these and other protocols are explained in more depth.

5.5.2 Benefits & Challenges

In the previous section, we elaborated on the key characteristics of Secure Multi-Party Computation to provide a foundational understanding of this PET. Now, we would like to first address the various advantages of Secure Multi-Party Computation before turning to the associated drawbacks.

Benefits

Privacy Privacy is the main reason why Secure Multi-Party Computation should be deployed in practice. There are many instances where it is advantageous to perform a computation together with other parties. However, trust in these parties is rarely guaranteed, so that possible cooperation often does not materialize. In this case, SMPC can help by enabling multiple parties to perform a joint computation without the need to disclose any of their sensitive data [107]. Therefore, the raw data doesn't have to be shared with a third party but remains securely in the hands of each participant [112]. So, a secure SMPC protocol ensures that each party receives only its own input and the collective result of the computation, but not any private information about its collaborators [116]. In contrast to Differential Privacy, there is also no trade-off between privacy and utility, which means that the SMPC process is able to produce highly accurate results without risking the privacy of the participants [112].

Legal & Regulatory Compliance By eliminating the need to share raw sensitive data, SMPC can also help companies comply with various laws and regulations. Although SMPC cannot be considered a silver bullet against any kind of privacy breach, it can certainly contribute to meeting parts of the most relevant regulations [109]. Strong security requirements also lead to more trust among parties that want to collaborate. As a result, many parties are willing to jointly perform computations that they wouldn't have done without Secure Multi-Party Computation. This facilitates cooperation and, thereby, the ability to leverage sensitive data from different sources to draw meaningful conclusions [117].

Quantum-Safe Security Protecting sensitive data is of paramount importance in a wide range of industries. However, with the recent emergence of quantum computing, data security is no longer necessarily guaranteed in many organizations. Many fear that their current safeguards may no longer be sufficient. This is where SMPC can provide a remedy. Depending on the chosen cryptographic primitives a protocol is based on, SMPC can be considered quantum safe. This means that companies can adopt SMPC without being concerned about future computations being compromised by quantum technology. [118]

Efficiency In addition, we would like to point out that SMPC is theoretically able to perform any distributed computation, meaning that any conceivable task can be securely computed [22]. However, this is not always practically feasible, as more complicated computations could lead to significant computational overhead. But unlike Homomorphic Encryption, SMPC can still be deemed efficient enough for many complex computations (e.g., comparisons), making

it suitable for numerous real-world use cases [82]. As a result, it is already considered to be *commercially ready* and has already been deployed to detect fraud and to predict heart disease in patients [109]. The applications of SMPC will be discussed in greater length in section 5.5.3.

Challenges

Although SMPC offers numerous benefits, it is also necessary to be aware of the challenges that must be faced when implementing SMPC.

Performance Overhead One of the drawbacks of Secure Multi-Party Computation is the introduced computational overhead. Even though in one of the last sections, we classified SMPC as *commercially ready*, this does not mean that it is an efficient technology for all types of calculations. In the case of computations over simple functions, for example, SMPC can provide a high level of efficiency [107]. However, for more complex computations that require - among others - extensive pre-processing of the data (e.g., creating the shares in Shamir's Secret Sharing), SMPC can require significantly more time compared to the same computations on raw data [119, 107]. The computational complexity can therefore be seen as heavily dependent on the type of calculation that has to be performed [107].

In contrast to Homomorphic Encryption, these computations are not carried out by a third party with extensive resources but must be performed between all participants. This leads to a potential problem. The computing power of some participating devices may not be designed for too overly complex calculations, making SMPC not suitable in this context [115]. Deciding whether SMPC is a viable option may therefore depend on the resource constraints of the participating devices. Finally, we would like to acknowledge that SMPC protocols have made great strides in terms of computational efficiency in the last decade [107]. This gives us hope that tasks which are too inefficient today will be feasible in the future.

High Communication Costs Another potential downside of SMPC is a higher communication overhead. Since the data cannot simply be passed to a third party who performs the desired computation, the parties must perform all the calculations among themselves. In order to do this, multiple communication rounds between the participants might be necessary [119]. This not only leads to longer computation times but also to higher communication costs for the network [112]. For instance, as explained in the last section, Shamir's Secret Sharing not only requires shares to be created but also to be distributed among the participants. This obviously requires much more communication between the different parties compared to performing the same computations on raw data. [109]

To conclude, organizations that decide to adopt Secure Multi-Party Computation should be aware of the increased communication burden that may be involved. However, there have already been several successful attempts to reduce the required communication [107]. So, although this issue is not yet resolved, we will most likely see further improvements in the future.

Trade-off between Security and Efficiency The trade-off between security and efficiency must also be taken into account [108]. As discussed in the last section, several types of adversaries might want to attack the protocol execution. If one wants to protect against any possible kind of attack from these adversaries, one can design a fully secure SMPC protocol that safeguards the private inputs of the participants [108]. However, these protocols tend to be a lot less efficient. Thus, one has to balance this trade-off based on the security and efficiency

requirements of a specific use case [107]. For instance, if one wants to implement a highly secure real-time application, Secure Multi-Party Computation might not be the best option.

To show the resulting higher computational overhead with increased security, we take the example of Yao's Garbled Circuits algorithm used by Bringer et al. [120]. Yao's algorithm can be deployed under the assumption of either semi-honest or malicious adversaries. However, when considering malicious attackers, the computation time is significantly longer. It can be shown that Yao's Garbled Circuits algorithm is ten times less efficient when assuming malicious adversaries in terms of computation time compared to the semi-honest security model. This demonstrates that if you want to increase security, you have to sacrifice computing power. [120]

High Level of Complexity Another major disadvantage of SMPC is its complexity. Although we were able to present the fundamental concepts of SMPC to a non-technical audience, the application in the real world is far from simple. Developing SMPC protocols is an obstacle for many companies that is difficult to overcome [121]. The reason for this is that all of these tasks require very domain-specific knowledge of Secure Multi-party Computation [122]. This leads to a high effort to build a fully functional SMPC scheme from scratch, which in turn is associated with high costs. [121].

To counteract this, several researchers have already taken it upon themselves to simplify the entire process to make SMPC more accessible to a non-expert audience [121]. For instance, Wirth et al. [123] developed EasySMPC, a tool that should help organizations adopt SMPC protocols. EasySMPC is a desktop application that allows its users to "securely sum up predefined sets of variables among different parties" [123]. The application provides a graphical user interface, and the users are not expected to write any code, thus making it a 'no-code' solution. To get a better insight into the specific applications of EasySMPC, we recommend reading the paper by Wirth et al. [123].

As you can see, research is already being conducted to overcome the complexity associated with the adoption of SMPC. However, organizations that want to leverage SMPC for various tasks today still have to deal with the complexity of designing and developing SMPC protocols [121].

5.5.3 Applications & Examples

The objective of the previous sections was to provide a fundamental overview of the characteristics, advantages, and disadvantages of Secure Multi-Party Computation. In this section, however, we want to address the question of where SMPC can be applied in practice.

Areas of Application

Privacy-Preserving Data Analytics One of the main application areas of Secure Multi-Party Computation is Privacy-Preserving Data Analytics [22]. As has been mentioned several times throughout this thesis, data analysis is of paramount importance in most industries. With its help, valuable insights can be extracted from the data, which can help companies to achieve their objectives. For instance, SMPC could be leveraged for statistical analyses across multiple organizations [22]. This could be used, for example, in the financial industry for proper risk assessment or fraud detection, but the potential application areas are vast.

Machine learning could also benefit from the introduction of Secure Multi-Party Computation. On the one hand, SMPC could enable the joint training of an ML model without the need for the collaborating parties to disclose their own private training data [107]. On the other

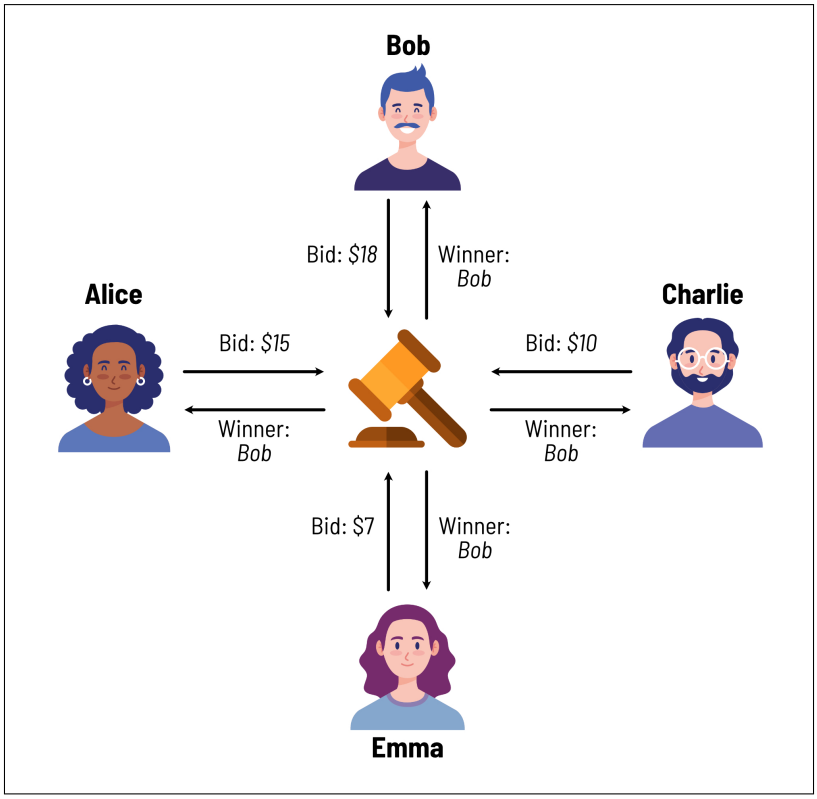


Figure 5.11: Sealed-Bid Auctions

hand, machine learning could also be performed via SMPC so that a party's private inputs are not revealed to the model owner [22]. As one can see, there are various ways in which SMPC can facilitate the private analysis of data. With further advances in this area, we can expect SMPC to be adopted by numerous companies in various industries to protect their sensitive data.

Secure Sealed-Bid Auctions Secondly, SMPC could help to conduct Secure Sealed-Bid Auctions. In a sealed-bid auction, several parties privately place their bids with the hope of acquiring a certain good or property (see Figure 5.11) [124]. The party who has placed the highest bid is the winner and thus also receives the item in question for the bid placed. Since the bids remain private and are not communicated to the other participants, it is not possible to adjust one's own offer based on the competitors' bids [125]. In sealed-bid auctions, care must be taken to ensure that no bids are compromised or even manipulated [107]. If this is not assured, the auction could become vulnerable to adversaries. For example, if a malicious party A knew the bid of its competitors, it could simply set its bid slightly higher than the competing bids and thus guarantee that it will win the auction [107]. To prevent this, SMPC can be leveraged to determine which party has placed the highest bid without all the privately placed bids becoming known to the other participants [106]. Due to the security guarantees of SMPC, one can also be assured that the protocol identifies the correct winner and the outcome cannot be influenced by adversaries [107]. For this reason, SMPC has already been deployed to ensure secure auctions in the real world. A suitable example is the famous sugar beets auction [126] that can be found in this section.

Privacy-Preserving Genomics Another application area in which SMPC has a promising future is Privacy-Preserving Genomics. As we have already shown with the example of the

SEBIS General Hospital, the analysis of genomes offers many advantages. However, due to its sensitive nature, the opportunities to conduct these kinds of analyses are rather limited. The reason for this is that most people are reluctant to share this type of information if their privacy cannot be guaranteed. This can lead to large-scale research studies not being carried out, although they would be of great importance to society. However, with the adoption of SMPC, multiple researchers can collaborate on such a study without sharing the patients' raw genome data. The results of these studies could then be used, for instance, to determine a patient's increased risk for a particular disease so that early measures can be taken and thereby possibly even saving lives. In summary, Privacy-Preserving Genomics could reach its full potential through the introduction of SMPC. [105, 104]

Ideal Use Cases

Even though Secure Multi-Party Computation is theoretically capable of performing any type of distributed computation, this does not mean it should be used for every problem. Hence, knowing which problems are ideally suited for SMPC is crucial. This is exactly what we will discuss in the following:

1. First, it is important to remember that SMPC can only be applied to problems where two or more parties want to cooperate. Since we have talked in this chapter repeatedly about calculations that are performed by several participants together, this should be self-evident. Moreover, the data has to be sensitive and is not allowed to be shared with the other participants. Otherwise, the different parties could simply pool their data and then perform the calculations even more efficiently. Furthermore, SMPC should only be employed if there is no external entity that all parties can trust. In fact, if the parties could decide to outsource the computation to a trusted third party, this would also be more efficient than SMPC.

2. Secondly, there should exist a reasonable trade-off between security and efficiency. The more you want to defend against semi-honest or malicious adversaries, the more you have to sacrifice computational power [107]. Striking the right balance between security and efficiency is unique to each application and therefore has to be decided on a case-by-case basis. For instance, if one wants to perform highly secure computations that are not time-critical, one can generally choose to employ SMPC. However, sometimes there is no reasonable trade-off. In this case, SMPC should not be deployed. For example, if one wants to implement a fully safe real-time application, one needs both security and efficiency. Therefore, it may not be appropriate to apply SMPC in this case.

3. Thirdly, SMPC should be applied when accurate results are needed [112]. In contrast to Differential Privacy, SMPC is able to perform calculations precisely while still protecting the private inputs of the participants. This is especially helpful in situations where approximate results are unacceptable. Fortunately, this doesn't come at the cost of secure computations, as one can still rely on the security guarantees provided by SMPC protocols [108]. For instance, if SMPC is used to compute accurate results, one can still be assured that a potential adversary cannot see the private inputs of the honest participants or even manipulate the output of the joint computation. Therefore, it can also be applied even if there is no trust between parties.

Primary Industries

Secure Multi-Party Computation has the potential to be used in virtually all industries in which joint computations have to be performed securely. Hence, its application is not limited to the private sector, as governmental institutions could also benefit from this PET [118].

Since SMPC facilitates data sharing and analyses across organizations, it is particularly beneficial in the healthcare and finance sectors [118]. Since the advantages of data sharing and analysis in both of these fields have already been discussed at length in the previous chapters, we will not repeat them here.

Maturity & Adoption

The idea of Secure Multi-Party Computation was first proposed in the 1980s by the Chinese scholar Andrew Yao [127]. At the time, however, it was far from being applied in practice, as it was very inefficient and offered only limited security guarantees. For the first 20 years after its inception, researchers were mainly concerned with its feasibility before then trying to improve its performance [22]. Not only through more efficient protocols but also due to improved hardware, SMPC was then finally ready to be applied in many use cases [121]. However, there are still some obstacles to overcome in implementing SMPC, which we discussed in detail in the last section.

Although we have labeled SMPC *commercially ready*, it is not yet widely used in the current industry [107]. Gartner has estimated in its published "Gartner Hype Cycle 2021" [44] that SMPC will still take another 5-10 years to reach the stage where it will be adopted by a large number of organizations [128]. However, SMPC can already be deployed today for a wide variety of use cases [121, 22]. Therefore, many organizations can already reap the benefits of adopting SMPC, even though there are still ongoing challenges to face.

Examples

A Preventable Privacy Accident In the previous chapters, we elaborated on companies that could have prevented a data breach if they had deployed a particular PET. But in this chapter, we would like to discuss an expensive accident that Secure Multi-Party Computation could have prevented.

In 2009, the US communications satellite *Iridium 33* accidentally collided with the Russian satellite *Cosmos 2251* [16, 129]. This incident resulted not only in significant financial losses but also in a large amount of debris that is still causing long-term problems in Earth's orbit, for example by leading to further collisions with other satellites.

This accident could have been prevented if the US and Russia had collaborated [16]. However, due to the lack of mutual trust between both parties, they wouldn't have been able to share their satellites' raw location data. But they could have employed Secure Multi-Party Computation instead. This would have enabled them to detect possible collisions without revealing any sensitive information [16]. Since then, several researchers have been exploring the idea of leveraging SMPC to detect satellite collisions [130, 131].

Successful Implementations To conclude this chapter, we would like to discuss two successful deployments of Secure Multi-Party Computation in the real world.

Several researchers from Boston University wanted to investigate the gender pay gap in Boston together with the Boston Women's Workforce Council. In order to obtain reliable results, they tried to collect the salaries of both men and women in various companies across different industries. However, here they encountered a problem. Due to privacy concerns,

the companies were unwilling to disclose their employees' salaries. But the researchers still managed to calculate the average salary difference between men and women with the help of the Secure Multi-Party Computation. By developing a web-based tool and making it available to the participating companies, the organizations were able to share their employees' salaries privately and securely. This allowed the necessary calculations to be performed without invading the employees' privacy. Thus, the researchers were able to leverage SMPC to determine the gender wage gap successfully. The results of this study were significant as they revealed large salary disparities between men and women. On average, a woman earns only $0.77 for every dollar a man makes. [118, 107]

In Denmark, there are several sugar beet farmers, but only one sugar beet processor called Danisco [126]. As a result, each sugar beet farmer has a contract with Danisco in which the quantity and price of the beets that can and must be delivered to Danisco are specified. Due to the change in market conditions in 2007/2008, these contracts had to be reallocated again. This reallocation was to take place by means of an auction [126]. The farmers, however, were worried that their bids could expose too much about their financial status [105]. Therefore, it was crucial for most farmers that their bids remain private and not be revealed to Danisco. In addition, it had to be ensured that a malicious attacker could not alter the final result, as this could, for example, lead to the wrong farmer being declared the auction winner. Therefore, the farmers joined forces with several scientists and the Danish government in order to build the first commercial application of Secure Multi-Party Computation [107]. This made it possible to identify a winner of the auction without revealing the bids of the participants [106]. If interested, we can refer to Bogetoft et al. [126], where this auction is explained in great detail.

5.6 Zero-Knowledge Proofs

Before we look at the definition and characteristics of Zero-Knowledge Proofs (ZKPs), we would first like to take another look at a problem that the SEBIS General Hospital is facing:

The SEBIS General Hospital would like to evaluate the data from a clinical trial. However, the analysis of the gathered data is too computationally intensive for their IT infrastructure. Therefore, they would like to transfer this sensitive data to an external party that can perform the calculation instead. Since this third party is not trustworthy, the SGH employs a PET (e.g., Differential Privacy or Homomorphic Encryption) to protect this data. However, there is no guarantee that the third party will perform the computation correctly. It is possible that the external vendor is malicious and intentionally provides an incorrect result. So, at first glance, it looks like they won't be able to outsource this computation.

However, they could deploy Zero-Knowledge Proofs to verify if the computation was performed correctly [132, 133]. In the following section, we would like to give an overview of the definition and characteristics of Zero-Knowledge Proofs in order to provide a foundational understanding of how this PET can be used to protect privacy.

5.6.1 Definition & Characteristics

Nowadays, individuals often find themselves in situations where they must convince others of a certain claim. However, because in most encounters, there is no mutual trust, one is usually obliged to prove the veracity of the statement. This often leads to the need to disclose sensitive data. For example, if you want to sign up for a digital trading platform, the service provider has to make sure that you are old enough to engage in financial trading. To prove this, you need to show your ID, which contains sensitive information such as date of birth or even place of residence. This leads to you having to sacrifice your privacy in order to be able to prove the statement, "I am over 18 years old". In certain situations, the stakes can be even higher, such as when health information must be disclosed. Zero-Knowledge Proofs have been developed to allow individuals to prove a certain claim without the need to sacrifice their privacy.

A Zero-Knowledge Proof allows one entity (the prover) to prove to another entity (the verifier) that a particular statement is true, whereby no information is disclosed to the verifier other than the fact that the prover's statement is true [132, 133]. The veracity of this statement can generally be proven by a secret that the prover possesses [23]. In an ideal world where there is mutual trust between the two parties, the prover could simply pass the secret on to the verifier, which can then check the truth of the statement independently. In this scenario, however, the secret is not kept confidential, which means that the privacy of the prover may be violated [134]. Therefore, in most real-world use cases, the secret information of the prover should not be disclosed to other parties. This may initially seem contradictory since the question arises as to how one can prove a statement without revealing any underlying information [135]. However, Zero-Knowledge Proofs were developed precisely for this purpose. To illustrate the concept of ZKPs, we would first like to provide a conceptual example before then addressing the key characteristics of Zero-Knowledge Proofs.

Example: Alibaba's Cave (Based on [136])

Many examples have been developed to introduce ZKPs to a broad range of audiences. In the following, we would like to focus our attention on the most prominent example, Alibaba's cave, as described in [136] (see Figure 5.12).

In this thought experiment, there are two parties, Alice (the prover) and Bob (the verifier). Alice claims to have the key to the door inside a cave and wants to prove this to Bob. The

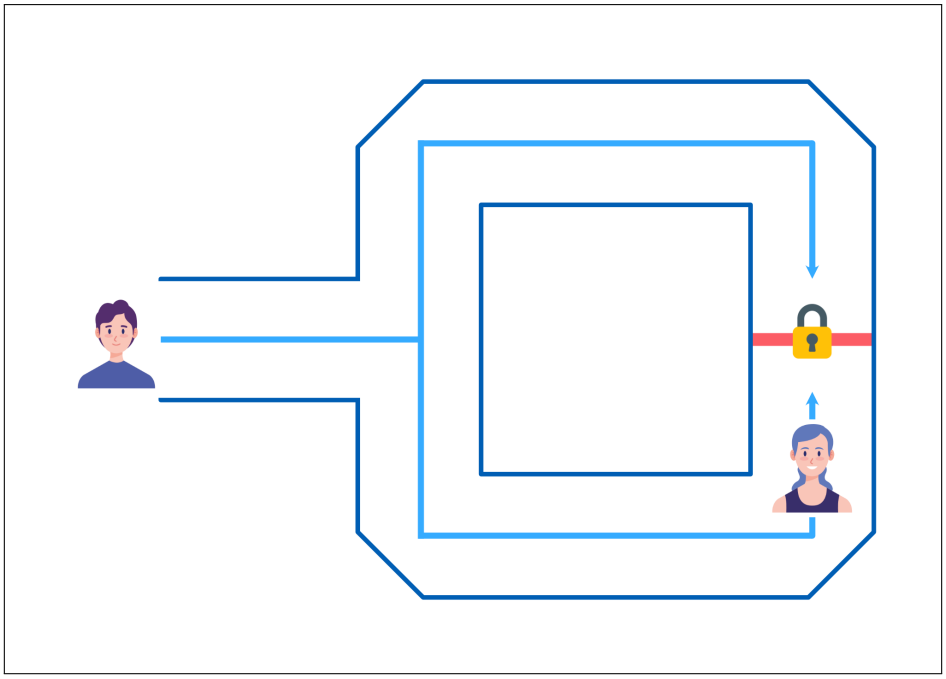


Figure 5.12: Alibaba's Cave (Based on: [136])

cave, as can be seen in Figure 5.12, has only one entrance but then splits into two paths that reconnect at a closed door. This door can only be opened by someone if they possess the corresponding key. To prove to Bob that she has this key, Alice proposes the following process:

1. Alice randomly chooses one of the two pathways without Bob's knowledge.
2. Bob then tells Alice which direction to come back out of the cave from.
3. Alice leaves the cave via the path chosen by Bob.

If these steps are taken, the following two results can occur:

If Alice does not have the key, she cannot fool Bob into thinking she has the right key. For example, if, by chance, she decides to enter the cave from the right, but Bob wants her to come out from the left, she will not be able to do so. This is because, without the key, she will not be able to get through the door and thus onto the opposite path.

However, if Alice has the correct key, then she can convince Bob by repeatedly following his instructions. That's because if Alice returns from the path specified by Bob the first time, he can be 50% sure that she has the right key. If she passes this test again, he can already be 75% sure. These steps can then be repeated until a specified accuracy is reached, thus fully convincing Bob.

ZKP Properties

There are various distinct implementations of Zero-Knowledge Proofs, each with its own specific use cases. However, they must all exhibit the following three properties to be considered a successful application of ZKP:

Completeness: If the statement to be proven is true, then at the end of the interaction, the prover has succeeded in convincing the verifier of the veracity of this specific statement [23].

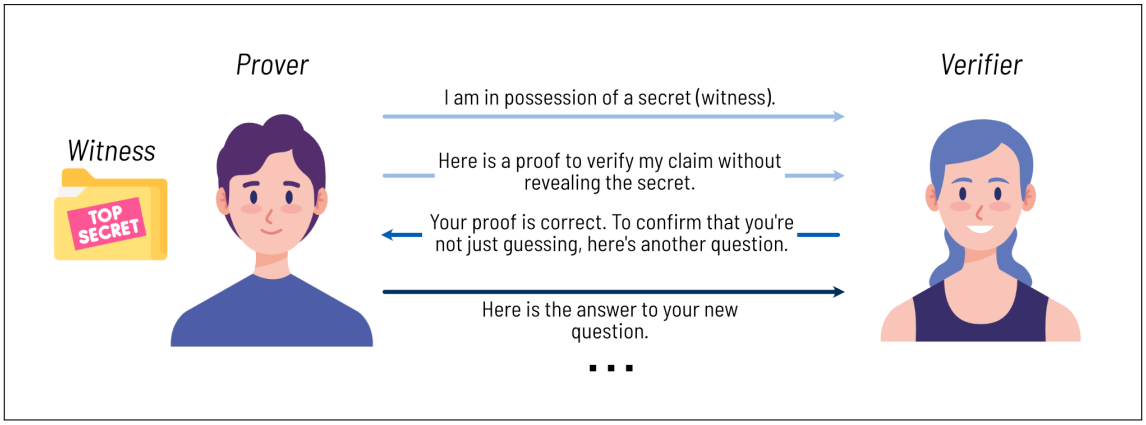


Figure 5.13: Visualization of the ZKP Process (Based on [133], [139])

Soundness: If the statement to be proven is false, a malicious prover is highly unlikely to successfully convince the verifier that the statement is true [137, 138].

Zero-knowledge: The prover does not disclose any sensitive information to the verifier other than the fact that the statement to be proven is true. This allows the prover to convince the verifier of the statement without compromising the confidentiality of the secret. [137]

ZKP Process

Having dealt with the definition and properties of Zero-Knowledge Proofs, we would now like to elaborate on the three main stages usually involved in proving ownership of a particular secret (see Figure 5.13):

Witness: The objective of the prover is to convince the verifier that he or she is in possession of a certain secret, also known as the witness to the proof. There are a set of questions that can only be answered by someone who has access to this secret. To prove possession of the secret, the prover randomly picks a question from this set and then calculates the corresponding proof. Finally, this proof is sent to the verifier. [133]

Challenge: Even if the prover answered the chosen question correctly, it still doesn't imply the possession of the entire secret. Therefore, the verifier challenges the prover by selecting a different question from the set and asking again for an appropriate answer. [133, 139]

Response: The prover receives the verifier's request and then calculates the correct response to the new question. Based on whether the prover's answer was satisfactory, the verifier will either accept or reject the proof. However, since the prover may have only guessed correctly, the verifier asks further questions to establish with a very high probability whether the prover knows the secret. [133, 139]

By following these steps, no sensitive information is revealed to the verifier, but the prover is still able to demonstrate the veracity of a specific statement.

Interactive vs Non-interactive ZKPs

The process described above is also called an Interactive Zero-Knowledge Proof because, in that instance, the prover and the verifier have to communicate several times. This poses some challenges. Since multiple rounds of interaction are required, a significant communication

overhead is introduced that can potentially make it unsuitable for many real-world use cases [139]. Both parties also need to be online and available throughout the entire ZKP process [133]. Moreover, Interactive ZKPs are not efficient when it comes to proving the same statement to several verifiers, as you have to convince each of them separately [135]. Non-interactive Zero-Knowledge Proofs have been developed to overcome these obstacles.

In Non-interactive ZKPs, there is only one interaction between the two parties. The prover sends only one proof to the verifier, who must then decide whether to accept or reject the claim [23]. To do this, both parties usually first exchange a piece of honestly generated public parameters [133, 140]. By using these public parameters combined with a particular algorithm, the prover can then compute a proof. Anyone who then has access to the parameters and this proof can check independently whether the statement to be proven is true or not. We want to note that in some cases, the computation of this kind of proof can be computationally intensive and may even be less efficient than the corresponding Interactive ZKP [23]. The transferability of the proof is a key advantage of Non-interactive ZKPs. This means that the same proof could be used to convince several verifiers at the same time [23]. As the proof can also be stored, the parties also do not need to be available simultaneously [135]. At this point, we do not want to explain how Non-interactive Zero-Knowledge Proofs work in detail, but we can refer to Blum et al. [140] for a more in-depth discussion.

ZKP Protocols

As already mentioned, there is not one true implementation of Zero-Knowledge Proofs, but rather a variety of ZKP protocols, each with its own use cases. Although we will not discuss them in detail, we would like to briefly introduce them below:

zk-SNARKs: Zero-Knowledge Succinct Non-interactive Arguments of Knowledge (zk-SNARKs) provide a method to create Zero-Knowledge Proofs that are both fast and efficient [134]. This is made possible by generating proofs that are small in size and can easily be verified [135]. This capability makes them particularly appealing for many real-world applications. ZK-Snarks are also considered a Non-interactive ZKP protocol, meaning there is only one interaction between the prover and the verifier. However, a significant disadvantage of zk-SNARKs is that the creation of the parameters must be done honestly by a trusted third party [141]. If the parameter generation is compromised, the security of the ZKP protocol is severely undermined.

zk-STARKs: In contrast to zk-SNARKs, Zero-Knowledge Scalable Transparent Arguments of Knowledge (zk-STARKs) are not dependent on a trusted third party generating the parameters needed for a Non-interactive ZKP [142]. As the name suggests, zk-STARKs can be considered scalable as they are significantly faster than zk-SNARKs for larger proof sizes [142]. Furthermore, they are regarded as more transparent because they don't rely on a third party for parameter generation but instead employ publicly verifiable randomness to determine the public parameters. However, longer proofs are created when employing zk-STARKs, which may cause an additional verification effort. In some cases, this may lead to zk-STARKs being slower than zk-SNARKs. [139]

Bulletproofs: Bulletproofs are another important protocol that can be used to implement Zero-Knowledge Proofs. Similar to zk-STARKs, Bulletproofs do not depend on a trusted third party to generate the public parameters [142]. In addition, the proofs generated in this protocol are usually very short [139]. In particular, they are used to create range proofs, i.e., they make it possible to verify whether a given piece of secret information is

within a certain range without revealing its exact value [142]. For example, Bulletproofs could be employed to prove that one is in their twenties without having to reveal their exact age. Because of that capability and its other beneficial properties, it is often used in cryptocurrencies to ensure secure transactions between parties [137]. Bulletproofs, however, are often very computationally intensive, resulting in higher verification times [139].

Although all of the above protocols have both advantages and disadvantages, a decision on which protocol to implement must be made on a case-by-case basis. Thus, each of the protocols has its raison d'être. Since a detailed discussion of these protocols is beyond the scope of this thesis, we would instead like to refer to Ben-Sasson et al. [143], Bünz et al. [144], and Bitansky et al. [145], where each of these protocols is explained in greater depth.

5.6.2 Benefits & Challenges

Following our explanation of the concept and features of Zero-Knowledge Proofs in the previous section, we would like to explore this PET's benefits and challenges. First, we will provide a brief overview of the main advantages of ZKPs, before then addressing the challenges in more detail.

Benefits

Privacy We would like to start with the most significant advantage of Zero-Knowledge Proofs, namely their ability to protect sensitive data and, thus, the privacy of the prover. As we mentioned in the last section, individuals often need to prove a certain claim to another party on a daily basis. By applying conventional methods, however, one has to disclose sensitive data to do this and thus sacrifice one's privacy. But with Zero-Knowledge Proofs, a party can prove the veracity of a particular statement without having to disclose any sensitive data [23]. Throughout the entire process, the provers' secret information remains secret as it is never shared with unauthorized third parties [146]. This results in anonymity for the prover and protection against possible man-in-the-middle attacks [138]. Because even if a malicious entity were able to intercept the messages exchanged between the prover and the verifier as a *man-in-the-middle*, the privacy of the prover would still be preserved as only a proof and not the secret itself is transmitted to the verifier [138, 146]. These properties are particularly beneficial in the financial sector, where one could adopt ZKPs to ensure secure transactions [147].

Legal & Regulatory Compliance Due to its zero-knowledge nature, ZKPs can also help companies to be compliant with privacy laws [147]. Typically, companies must gather a large amount of information about their customers (e.g., biometric data for authentication purposes) and then protect that sensitive data according to several privacy regulations [138]. However, Zero-Knowledge Proofs allow companies to verify certain claims, such as the identity of a customer, without the need to collect the corresponding sensitive data [148]. In some cases, this could lead to a mitigation of data breaches, as there would be no data to breach. Furthermore, companies could also leverage ZKPs to prove a certain claim themselves without handing over their sensitive data. Therefore, Zero-Knowledge Proofs can support companies in achieving compliance with portions of the most relevant data protection laws [136]. However, we would like to emphasize that ZKPs are not a universal solution to all data protection-related problems an organization may face.

Efficiency Zero-Knowledge Proofs can also be considered efficient, depending on the type of ZKP implementation used. In many traditional instances, the prover and the verifier would need to exchange a large amount of confidential information in order to verify the correctness of a statement [149]. However, with the help of ZKPs, only a proof needs to be transmitted to the verifier, potentially reducing network traffic [138, 149]. However, ZKPs can also be computationally intensive. Multiple rounds of communication and complex calculations for proving and verifying a particular statement may slow down the process, depending on the specific ZKP implementation [23, 150]. Although we don't want to go into detail here, this will be discussed in greater length in the next subsection.

Challenges

As we illustrated in the last subsection, Zero-Knowledge Proofs have many advantages. However, there are also some challenges that companies that want to adopt this PET need to overcome.

Negative Influences on Efficiency & High Operational Cost Although Zero-Knowledge Proofs are generally considered efficient, their performance heavily depends on the specific protocols and use cases. The question now arises of how to measure its performance and what factors might influence the efficiency of a ZKP process. In the following, our discussion is based on the paper by Benarroch et al. [23].

A key metric is the number of communication rounds, i.e., how many messages must be exchanged between the prover and the verifier to complete the ZKP process [150]. Interactive ZKP protocols often require several interactions between the two parties before the verifier is convinced of the prover's statement. This can take a lot of time and thus decrease the efficiency of the ZKP process. It is also essential to consider the size of the messages that the prover and verifier need to exchange [23]. The larger the proofs, the longer it takes to transfer them from the prover to the verifier. In addition, the computations for the proof and verification tasks must also be taken into account [151]. Depending on how complex the operations involved with these tasks are, the entire ZKP process can become very inefficient in some circumstances. There may even be certain set-up computations that need to be performed [23]. For example, Non-interactive ZKPs require public parameters to be generated at the very beginning, which also adds time to the process.

Furthermore, costs can potentially increase due to the complex operations necessary to create and verify Zero-Knowledge Proofs. For example, better hardware may be required to perform the calculations efficiently. This would entail new investments that can potentially be very expensive. Communication costs may also increase if the number of communication rounds is too high or the proof is large in size. It should also be borne in mind that too complex ZKP operations are not feasible for most individuals' devices. Therefore, Zero-Knowledge Proofs could also lead to an increase in costs for end users. [133, 139]

Finally, we would like to emphasize again that the performance and cost of Zero-Knowledge Proofs are highly dependent on the applications and selected protocols. This, in turn, has implications for the scalability of ZKPs [146]. Therefore, it is not possible to make a universal statement about the efficiency of ZKPs.

Non-Deterministic Another challenge we would like to address is the fact that Zero-Knowledge Proofs are more probabilistic than deterministic [152]. This implies that Zero-Knowledge Proofs are only able to verify a particular statement with a certain probability. So there is no ZKP protocol that can assure with 100 percent probability that the prover's

statement is actually truthful [150]. To be truly confident, the verifier would need to have access to the underlying data. However, since this data is a secret and should remain confidential, it cannot be disclosed to unauthorized entities. However, we would like to emphasize that this does not mean that ZKPs are inherently untrustworthy. The Zero-Knowledge Proof process can be constructed in such a way that a particular statement can be regarded as true beyond any reasonable doubt [142]. For example, with Interactive ZKPs, several rounds of communication must simply be carried out until the verifier is convinced [150]. In the Alibaba's Cave example we discussed in the last section, after 20 rounds of communication, Bob could be 99.9999% certain that Alice is in possession of the correct key.

Quantum Resistance In this thesis, we have already discussed several times the ability of certain PETs to withstand quantum attacks. Even though quantum computing is only in its nascent stages and hasn't any major security implications today, many companies still have to ask themselves whether their systems will continue to be secure in the future. In contrast to Homomorphic Encryption, many ZKP protocols provide no protection against quantum attacks [138, 133]. For instance, both zk-SNARKs and Bulletproofs are based on mathematical assumptions that are assumed to provide no quantum resistance. However, zk-STARKs are deemed secure against quantum attacks, and additionally, significant progress is already being made to make ZKP more quantum-safe in general. [133, 153]

Lack of Standardization Finally, the prevailing lack of standardization is another challenge for Zero-Knowledge Proofs. As it stands, there are no international standards or guidelines to follow. This could make the adoption of Zero-Knowledge Proofs much more challenging, as implementing ZKPs is not a simple task. Until there are universally accepted international standards for this PET, widespread adoption is unlikely [146]. Despite these challenges, we would like to point out that there already are community references that can provide guidance [23]. Additionally, the International Organization for Standardization is currently working on "Guidelines on privacy preservation based on zero knowledge proofs" [154].

5.6.3 Applications & Examples

In the previous sections, we have aimed to provide a non-technical audience with a basic understanding of Zero-Knowledge Proofs. In this section, we would like to elaborate on this by answering the question of where ZKPs can be employed in practice.

Areas of Application

Anonymous Authentication Anonymous Authentication enables an individual to authenticate themselves without revealing anything about their identity [155]. Authentication often involves either a password or biometric data (e.g., fingerprints, facial recognition, etc.). As this is confidential information, it should not be shared with an untrusted third party. With the help of ZKP, one can authenticate without sharing this information with the verifier [148].

For example, if an individual wants to log in to their bank, instead of a password, they could present a Zero-Knolwedge Proof that verifies they have access to the password without actually revealing it [152]. After the bank has verified the ZKP, it grants the person access to the corresponding bank account without ever having received the actual password. This means that even if the database is breached, no sensitive information can be stolen, as the bank does not even possess this information [152, 148]. Explaining exactly how Anonymous

Authentication works is outside the scope of this work. However, we can refer to Zhang et al. [155], who discuss the implementation in detail.

Anonymous Payments Anonymous Payments are another major application of Zero-Knowledge Proofs. In most financial transactions, the identity of the sender and the recipient, as well as the type of asset and the transaction amount, must be disclosed to a financial institution [23]. As this is sensitive information, it should ideally not be disclosed to third parties. To achieve this, ZKP can be deployed to ensure the privacy of the participants during such a transaction. For instance, one could validate whether an individual has enough funds to carry out a certain transaction without revealing anything else about that person [137].

In particular, ZKPs can be leveraged for more private transactions in the blockchain. The most predominant cryptocurrencies, Bitcoin and Ethereum, are both public blockchains, which means that all transactions are publicly verifiable [150]. This is great for transparency but detrimental to the privacy of users, as the addresses of the sender and recipient, the transaction amount and the timestamp of the transaction are publicly available. However, ZKPs have already contributed to making blockchains more private by hiding the transaction details such as addresses, asset types, amounts, and timelines [133]. Zcash is an already existing implementation of Anonymous Payments in the blockchain that we will examine later in this section.

Verifiable Computations Another crucial application of Zero-Knowledge Proofs is Verifiable Computations. Nowadays, it is common to delegate certain computational tasks to third parties, often because one does not have enough computing power. However, if the party is malicious, they might try to deceive you by providing a wrong answer. To prevent this from happening, Zero-Knowledge Proofs can be employed to verify the correctness of such a calculation [133].

As discussed in the penultimate chapter, Homomorphic Encryption can guarantee the confidentiality of data but not its integrity [88]. The party performing the operations on the encrypted data might not follow the instructions and deliberately give you the wrong answer. However, with the help of Zero-Knowledge Proofs, it would be possible to guarantee the integrity of the data by the party performing the calculations transferring a proof that verifies that the calculation was executed correctly [133]. At this point, however, we would like to emphasize that the use of ZKPs produces additional computational overhead. So if verifying the proof takes longer than computing the outsourced computation, ZKP should not be applied [88].

Ideal Use Cases

Although Zero-Knowledge Proofs are a great solution to a number of challenges, they are not suitable for all privacy-related problems. Therefore, in the following, we would like to explore the question of which characteristics of a problem suggest that a Zero-Knowledge Proof is the right technology to address it:

1. First, Zero-Knowledge Proofs are only helpful in scenarios where a certain statement must be verified. However, the only way to prove the veracity of this statement is through access to specific sensitive data that only the prover knows. Since this sensitive information is not to be shared, verification must take place in such a way that the verifier does not find out any further information about the prover's secret information. [23]

2. Moreover, Zero-Knowledge Proofs can be applied even if there is no trust between the participating entities. Due to its zero-knowledge nature, no useful information is revealed to the verifier throughout the entire ZKP process, aside from whether the statement is true or false [146, 141]. For example, one can outsource computations to a cloud service provider and request proof that the calculations were performed correctly. This allows two parties who do not trust each other to communicate in a secure and private manner. [88]

3. Although Zero-Knowledge Proofs are efficient enough for many use cases, they must not be used for overly complex proofs. In some scenarios, the time involved in creating and verifying a proof is too high [151]. If this is the case, ZKP may not be applicable. Additionally, we would like to point out that the various ZKP protocols have distinct use cases. For instance, while zk-SNARKs are applied for smaller proof sizes, zk-STARKs are more scalable and can thereby handle larger proofs as well [142]. Hence, whether ZKP is efficient depends heavily on the particular protocol and use case.

Primary Industries

As with other Privacy-Enhancing Technologies, Zero-Knowledge Proofs are not limited to a single industry or application area. After all, the verification of claims being made by a particular party is essential in any field. Be it to authenticate a person or maybe even to verify a financial transaction between two entities.

The financial sector, in particular, can benefit from Zero-Knowledge Proofs. As mentioned earlier, most transactions involve providing a great deal of sensitive data to a financial institution (e.g., type of assets, transaction amount, etc.) [23]. However, with Zero-Knowledge Proofs, it would be possible to enforce anonymous transactions. Another relevant application area of Zero-Knowledge Proofs is the healthcare sector, as it often involves sensitive data that should not be disclosed. For example, it would be possible to prove that one has access to a patient's medical records without disclosing them [156]. Also, in the case of outsourced computations, ZKP could help hospitals to ensure that the third party does not violate the integrity of the data by providing wrong answers [88].

The possible application areas of Zero-Knowledge Proofs are vast. This PET can help both companies and private individuals to protect their privacy.

Maturity & Adoption

The concept of Zero-Knowledge Proofs was first proposed in 1985 in the paper "The knowledge complexity of interactive proof systems" by Shafi Goldwasser, Silvio Micali, and Charles Rackoff [132]. Since then, there have been major advancements in this area, not only in terms of efficiency but also with regard to the development of new protocols with different use cases [23] (e.g. the introduction of Non-interactive ZKPs [157]). Nowadays, Zero-Knowledge Proofs can already be applied in practice, and many industries could benefit from their adoption [23].

Regarding the maturity and acceptance of Zero-Knowledge Proofs in the current industry, the same trend can be observed as with the other Privacy-Enhancing Technologies. According to Gartner's "Hype Cycle For Digital Identity 2022", a large number of companies have not yet implemented Zero-Knowledge Proofs in their business, and widespread adoption is estimated to take another 5-10 years [158]. Although Zero-Knowledge Proofs have their drawbacks, they can still be implemented today by organizations that value the benefits of being an early adopter.

Examples

A Preventable Privacy Breach First, we would like to have a closer look at an example of a privacy breach that could have been prevented if Zero-Knowledge Proofs had been employed.

The largest professional networking platform, LinkedIn, experienced a massive data breach in 2012, in which the data of more than 100 million users was compromised. The leaked data included not only the email addresses of these users but also their encrypted passwords used for authentication. However, as LinkedIn did not take the proper precautionary measures, these passwords could be cracked. Clearly, this represents a major breach of privacy for LinkedIn users. [159]

LinkedIn could have employed Zero-Knowledge Proofs to enable Anonymous Authentication. With this method, they could have verified the individuals' identities without storing their sensitive data in a central database. Thus, even if the attackers had gained access to the LinkedIn database, they would not have been able to obtain any useful information. [152, 148]

Successful Implementations Finally, we would like to present two successful implementations of Zero-Knowledge Proofs in the real world.

As discussed in detail in this section, most financial transactions entail handing over a significant amount of sensitive data to a financial institution [23]. This has serious privacy implications. A commonly held belief is that all cryptocurrencies are not only secure but also address these privacy concerns by being completely anonymous. However, most cryptocurrencies (e.g., Bitcoin or Ethereum) are public blockchains, so the transaction data is publicly available [150]. As the sender/recipient address doesn't directly reveal the user's true identity, Bitcoin can be considered pseudonymous [160]. However, by analyzing the transaction data of a specific address, it still might be possible to link it back to a real identity [160]. For this reason, public blockchains can still be considered a privacy concern. Z-Cash was developed to enable anonymous transactions in the blockchain and thereby protect the privacy of its users. It is a cryptocurrency based on Bitcoin's code, but it offers significantly higher privacy guarantees by leveraging Zero-Knowledge Proofs [134]. Instead of publicly available, the transaction data is encrypted [152]. This obscures the transaction amount and even the address of the sender and the recipient [137]. However, the transactions must still be checked for validity. For instance, a malicious actor could try to sell more coins than they currently own. Therefore, Zero-Knowledge Proofs, more specifically zk-SNARKs, have to be employed to confirm the validity of the transactions without revealing any private transaction data. [137, 152]

In today's digital world, you have to authenticate yourself to access any kind of service or platform on a daily basis. From social media platforms to more sensitive services such as online banking, it is necessary to ensure that only the authorized person has access to it [133]. To prove one's identity, people are usually compelled to disclose sensitive information. However, Zero-Knowledge Proofs can circumvent this by allowing for Anonymous Authentication. Keyless Technologies is a cybersecurity company that focuses on secure passwordless authentication methods. They leverage *Zero-Knowledge Biometric Authentication* to protect their users' privacy. This means that Keyless uses biometric data - such as fingerprints and facial recognition - for authentication without compromising the privacy of its users. Instead of sending the actual biometric data to Keyless, users can simply generate a Zero-Knowledge Proof and submit it to Keyless, which will then attempt to verify the user's identity. That way, an individual can prove they have the correct biometric template without actually disclosing it to Keyless. The big advantage of Anonymous Authentication is that even if a malicious party gains access to Keyless' network, they cannot obtain any sensitive information, as the biometric data has not

even been shared with Keyless. To conclude, we would like to point out that this method also complies with various data protection laws and regulations, such as the GDPR. [148]

6 Discussion

As mentioned at the beginning of this thesis, there is currently a lack of understanding and, therefore, a lack of adoption of Privacy-Enhancing Technologies in the industry. However, the significance of PETs should not be underestimated. They can help companies comply with relevant privacy regulations and protect individuals from potential privacy breaches. Now the question arises as to why many organizations are reluctant to adopt Privacy-Enhancing Technologies. One of the reasons is the current lack of educational materials that explain PETs in a simplified manner. Appropriate learning content would make the subject more accessible to non-technical audiences, such as managers responsible for deciding whether to adopt these technologies.

While a substantial body of literature already attempts to provide an overview of Privacy-Enhancing Technologies, most publications are only directed at experts and practitioners. Furthermore, existing literature targeted to a non-technical audience is either oversimplified or only discusses a single PET in detail without providing an overview of the PET landscape. The goal of this thesis is to fill the gap in current research by offering a comprehensive introduction and overview of the most prevalent PETs for data processing and analysis, tailored to a non-technical readership. This includes information about the characteristics, benefits, challenges, and potential use cases of the leading Privacy-Enhancing Technologies: Federated Learning, Differential Privacy, Homomorphic Encryption, Secure Multi-Party Computation, and Zero-Knowledge Proofs. To accomplish this, we developed educational material that introduces this topic to an audience without prior knowledge in this field.

Key Results and Implications

The created learning content consists of one learning path for each identified PETs. Each learning path is comprised of three learning nuggets: **Definition & Characteristics**, **Benefits & Challenges**, and **Applications & Examples**. These Learning Nuggets can be used to gain a basic understanding of the fundamental aspects of each technology. In addition, the accompanying quiz can be used for better knowledge retention and for self-assessment purposes.

The learning content created as part of this work can be used to educate a diverse group of people about the most prevalent PETs. It can be used, for example, by lawyers who need to ensure compliance with crucial privacy laws or by policymakers who can effect positive change in legislation. In addition, it can also benefit managers not involved in implementation but who simply want to get an overview of which technologies can be applied for which purposes. As you can see, our work is suitable for a wide range of audiences who do not need detailed implementation instructions but merely want to gain a foundational understanding of this subject. Therefore, we hope that this thesis, along with future work in this field, will raise awareness of Privacy-Enhancing Technologies and contribute to their widespread adoption.

6.1 Limitations

Having discussed how our work can benefit a wide range of people, we would now like to address the limitations of our work.

First, we mainly focused on PETs that can be applied for data processing and analysis. However, sensitive data must be protected during all stages of the data lifecycle (e.g., data transfer, storage, etc.). Therefore, organizations may need to consider additional PETs for an all-encompassing privacy protection strategy. Furthermore, we have only covered the five most prevalent PETs in the current literature. However, there are many other Privacy-Enhancing Technologies that can also be applied to ensure the protection of sensitive data. So even if none of the technologies mentioned in this thesis are useful for an organization, that does not mean that there are no other, lesser-known PETs that could serve their business. In addition, since this overview is intended to serve a broad audience, we could not tailor it to a specific target group. That is why we were unable to provide guidance specific to particular industries or contexts.

Moreover, this thesis does not provide any instructions on how to implement or deploy the various Privacy-Enhancing Technologies discussed in this work. Thus, although our thesis serves as a good introduction to the topic, if one wants to adopt these technologies, one must consult other, more comprehensive literature. Finally, we would like to point out that this work does not provide any specific legal guidance regarding the use of the identified technologies to comply with privacy laws and regulations. However, its significance should not be underestimated and should definitely be considered before adopting any of the aforementioned technologies.

6.2 Future Work

Even though our thesis represents a step in the right direction, there continues to be much that needs to be done in order to achieve the widespread adoption of PETs. Therefore we would like to briefly discuss possible future work in this area.

This thesis focused only on conveying information to a non-technical audience and therefore covered only the elementary levels of Bloom's Taxonomy. However, there is also a need for a more sophisticated level of education and training. Thus, creating more advanced educational materials that address more dimensions of Bloom's Taxonomy would help achieve widespread adoption. Moreover, this thesis focuses exclusively on the five most predominant PETs in the current literature. However, many other PETs can help organizations protect privacy and comply with relevant laws and regulations. Therefore, there would be great value in creating similar learning content for other PETs.

Implementing Privacy-Enhancing Technologies is usually a complex process and hence a timely and costly endeavor. This is one of the main reasons many organizations are reluctant to adopt them. The complexity of introducing these technologies into an organization could be reduced by providing a guide for implementing and deploying these PETs. This would help organizations integrate these technologies into their business, which in turn could lead to the greater adoption of PETs.

Next, the effectiveness of the learning content must be evaluated. Even though it has been developed with a non-technical audience in mind, it has yet to be proven that it actually serves its purpose. Moreover, further teaching and learning activities can be derived on the basis of this thesis. This may include lectures, tutorials, project works, or even workshops [13]. These can then be integrated into various learning platforms to disseminate the content of this thesis to a broader audience.

Finally, we would like to point out that the Privacy-Enhancing Technologies covered in this thesis are evolving at a rapid pace. Therefore, it is necessary to ensure that the developed learning content is kept up to date and adjusted to new developments.

7 Conclusion

Privacy-Enhancing Technologies have been developed to address growing concerns about the invasion of privacy. Furthermore, they help organizations in various industries comply with crucial privacy laws and regulations. However, due to the large volume, complexity, and heterogeneity of available information about them, much work remains to be done to achieve their widespread adoption. This thesis represents a step in the right direction by providing a comprehensive overview of the most prevalent Privacy-Enhancing Technologies for a non-technical readership. The learning content developed as part of this work can therefore be useful to a wide range of audiences.

Integral to the development of this thesis were the research questions formulated in section 4.1. In the following, we would like to revisit these questions and briefly answer each of them:

RQ1: What are the most prevalent PETs for data processing and analysis?

The goal of the first research question was to identify the most predominant PETs for data processing and analysis. A PET was considered more predominant than another if it had more publications in the current literature. The data needed to answer this question was obtained through a quantitative analysis. First, we compiled a list of PETs to be included in the analysis based on the comprehensive literature review conducted by Garrido et al. [24]. Next, we searched for each PET in various academic databases: IEEE Xplore, Scopus, ACM Digital Library, and ScienceDirect. We then counted the number of search results for each PET across the different databases to determine how relevant they were overall. Finally, since there was some overlap between the databases in terms of duplicate publications, we had to deduplicate the entries by title. Below are the final results of the quantitative analysis and hence the answer to which PETs are the most prevalent:

1. Federated Learning
2. Differential Privacy
3. Homomorphic Encryption
4. Secure Multi-Party Computation
5. Zero-Knowledge Proofs

RQ2: What are the characteristics, benefits, challenges, and applications of the selected PETs?

After determining which Privacy-Enhancing Technologies are most predominant, we addressed the second research question, the goal of which was to obtain more information about their characteristics, benefits, challenges, and applications. The first step in this process involved identifying the relevant publications from a large body of work. This was achieved by conducting a Systematic Literature Review, in which we searched for each PET in several academic databases with the help of search strings. Using a central database to collect the information and color coding to highlight the most critical information, we then extracted

the relevant data for our work. Finally, we synthesized the collected information, which then formed the basis for the development of the learning content. In this process, additional gray literature, such as blog posts and magazines, was used to supplement the knowledge gained from the SLR. A brief overview of the synthesized results of this research question is provided in Table 7.1.

RQ3: How can one convey this information in a meaningful and engaging manner to ensure a better understanding of the selected PETs?

The final research question is about how we can now convey the collected information in a meaningful and engaging way to ensure a better understanding of the identified PETs. We were able to accomplish this through the use of several theoretical learning frameworks, such as Bloom's Revised Taxonomy [14] and Gagné's Nine Events of Instruction [15]. While Bloom's Revised Taxonomy was leveraged to articulate the learning objectives we want to achieve, Gagné's Nine Events of Instruction helped us with the design of the learning content by providing advice on how to make it more engaging.

These learning frameworks were used to create an optimal learning environment for the developed educational materials. Furthermore, an accompanying quiz was designed to enhance the learner's retention. All of the created learning content can be downloaded as a PDF file at this link:

https://1drv.ms/f/s!Ag9Y6xObvis1ry2mvE58uj05X8fj?e=Scj1uX

By answering all the research questions mentioned above, we have succeeded in creating engaging educational materials for a non-technical audience. Together with the quiz, this learning content can now educate a diverse audience on the fundamental principles of the most predominant PETs. We hope our work will raise more awareness about the promising future of Privacy-Enhancing Technologies and that it can be considered a valuable contribution to making PETs more widely accessible.

Table 7.1: Systematic Literature Review - Summarized Results

	Federated Learning	**Differential Privacy**	**Homomorphic Encryption**	**Secure Multi-Party Computation**	**Zero-Knowledge Proofs**
Definition & Characteristics	Federated Learning allows ML models to be trained without the need to store raw data in a central location. This is achieved by shifting the responsibility of training the model from the central server to the clients. [19, 32]	Differential Privacy is a rigorous mathematical definition of a privacy goal. It can be accomplished by adding random noise to raw data in order to protect the sensitive information of the individual participants. [47]	Homomorphic Encryption is an encryption scheme in which the data remains encrypted even during processing. This means that a third party who is not authorized to read the data can still perform useful operations on the ciphertext. [76]	Secure Multi-Party Computation enables multiple parties to perform a joint computation without the need to reveal any of their private inputs in the process. [107]	Zero-Knowledge Proofs allow one entity (the prover) to prove to another entity (the verifier) that a particular statement is true, whereby no information is disclosed to the verifier other than the fact that the prover's statement is true. [132, 133]
Benefits	Privacy, Security, Legal & Regulatory Compliance, Improved Model Performance & Predictions, Cost Reduction, Scalability	Privacy, Security, Legal & Regulatory Compliance, Versatility, Customizability, Accessibility, Lightweight Computations, Data Preservation	Privacy, Security, Legal & Regulatory Compliance, Secure Outsourcing of Computation	Privacy, Legal & Regulatory Compliance, Efficiency, Quantum-Safe	Privacy, Legal & Regulatory Compliance, Efficiency
Challenges	High Communication Cost, Device Heterogeneity, Data Heterogeneity, Privacy Threats, Security Threats	The Optimal Trade-Off between Utility & Privacy, Privacy Budget Allocation, High Level of Complexity, Lack of Tools & Standards, Necessity for Change, Small Datasets, No Individual Level of Analysis	Performance Overhead, High Level of Complexity, Scalability Barriers, Constraints on Operations	Performance Overhead, High Level of Complexity, High Communication Cost, Trade-Off between Security and Efficiency	Negative Influences on Efficiency, Non-Deterministic, Quantum Resistance, Lack of Standardization, High Operational Cost
Applications	FL can be used with: Smartphones, IoT Devices, Organizations	Statistical Analysis, Machine Learning	Secure Cloud Computation, Secure Machine Learning, Supply Chain Security	Data Analytics, Genetic Testing, Secure Auctions	Anonymous Authentication, Anonymous Payments, Verifiable Computations
Examples	Ai.type*, Google GBoard, Apple Siri	Group Insurance Commission*, US Census, Apple	Cambridge Analytica*, Banco Bradesco S.A. (Pilot), Facebook (Hypothetical)	US Communications Satellite Iridium 33*, Boston Gender Wage Gap, Danish Sugar Beet Farmers	LinkedIn*, Keyless Technologies, Z-Cash

* *Example of a preventable privacy breach or an expensive accident.*

List of Figures

List of Tables

List of Abbreviations

Laws and Regulations

- **GDPR** General Data Protection Regulation
- **CCPA** California Consumer Privacy Act
- **HIPAA** Health Insurance Portability and Accountability Act

Federated Learning

- **FL** Federated Learning
- **HFL** Horizontal Federated Learning
- **VFL** Vertical Federated Learning
- **FTL** Federated Transfer Learning

Differential Privacy

- **DP** Differential Privacy
- **LDP** Local Differential Privacy
- **GDP** Global Differential Privacy

Homomorphic Encryption

- **HE** Homomorphic Encryption
- **PHE** Partial Homomorphic Encryption
- **SWHE** Somewhat Homomorphic Encryption
- **FHE** Fully Homomorphic Encryption

Secure Multi-Party Computation

- **SMPC** Secure Multi-Party Computation
- **SSS** Shamir's Secret Sharing
- **OT** Oblivious Transfer

Zero-Knowledge Proofs

- **ZKP** Zero-Knowledge Proof
- **SNARK** Succinct Non-interactive Arguments of Knowledge
- **STARK** Scalable Transparent Arguments of Knowledge

Other

- **PET** Privacy-Enhancing Technology
- **ML** Machine Learning
- **SLR** Systematic Literature Review
- **SGH** SEBIS General Hospital
- **GWAS** Genome-Wide Association Study

Bibliography

[1] M. Tom. *Root Insurance Leverages ML and Amazon SageMaker to Offer Fair Insurance Rates*. URL: https://aws.amazon.com/blogs/startups/root-insurance-leverages-sagemaker-to-offer-fair-insurance-rates/ (visited on 08/08/2023).

[2] C. A. Gomez-Uribe and N. Hunt. "The Netflix Recommender System: Algorithms, Business Value, and Innovation". In: *ACM Trans. Manage. Inf. Syst.* 6.4 (2016). ISSN: 2158-656X. DOI: 10.1145/2843948.

[3] A. Lukács. "What is Privacy? The History and Definition of Privacy". In: 2016. URL: https://api.semanticscholar.org/CorpusID:190478776.

[4] S. D. Warren and L. D. Brandeis. "The Right to Privacy". In: *Harvard Law Review* 4.5 (1890), pp. 193–220. ISSN: 0017811X. URL: http://www.jstor.org/stable/1321160 (visited on 08/05/2023).

[5] K. Renaud and D. Gálvez-Cruz. "Privacy: Aspects, definitions and a multi-faceted privacy preservation approach". In: *2010 Information Security for South Africa*. IEEE. 2010, pp. 1–8. DOI: 10.1109/ISSA.2010.5588297.

[6] C. Adams. *Introduction to Privacy Enhancing Technologies: A Classification-Based Approach to Understanding PETs*. Springer International Publishing, 2021. ISBN: 9783030810429. DOI: 10.1007/978-3-030-81043-6.

[7] Information Commissioner's Office. *Legal definitions*. URL: https://ico.org.uk/for-organisations/data-protection-fee/legal-definitions-fees/ (visited on 08/05/2023).

[8] General Data Protection Regulation (GDPR). *Article 1 – Subject-matter and objectives*. 2016. URL: https://gdpr-info.eu/art-1-gdpr/ (visited on 07/02/2023).

[9] State of California Department of Justice, Office of the Attorney General. *California Consumer Privacy Act (CCPA)*. 2023. URL: https://oag.ca.gov/privacy/ccpa (visited on 07/02/2023).

[10] The Centers for Disease Control and Prevention (CDC). *Health Insurance Portability and Accountability Act of 1996 (HIPAA)*. 2022. URL: https://www.cdc.gov/phlp/publications/topic/hipaa.html#print (visited on 07/02/2023).

[11] Information Commissioner's Office. *Privacy-enhancing technologies (PETs) - Draft anonymisation, pseudonymisation and privacy enhancing technologies guidance*. 2022. URL: https://ico.org.uk/about-the-ico/media-centre/news-and-blogs/2022/09/ico-publishes-guidance-on-privacy-enhancing-technologies/.

[12] University of Twente. *Examination & assessment – a step-by-step guide*. Revised version of the original Step-by-step guide for CW/CS + PSY, University of Twente. Reconstructed October 2015 for the faculty BMS. English version: Nov. 2016. Support: CELT. 2016. URL: https://www.utwente.nl/en/examination/manuals-examination/step-by-step-guide-examination-revised-jan2017.pdf.

[13] J. Dijkstra, S. Gerhards, M. Krooi, M. Menten, E. Radulova, M. Spigt, R. Vaatstra, and P. Vermeer. *The UM Handbook for Constructive Alignment*. Maastricht University Institute for Education Innovation (EDLAB), 2017. URL: https://edlab.nl/wp-content/uploads/2022/01/CoAl_PDF_final_version.pdf.

[14] R. Heer. *A Model of Learning Objectives–based on A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. URL: https://www.celt.iastate.edu/wp-content/uploads/2015/09/RevisedBloomsHandout-1.pdf.

[15] Northern Illinois University Center for Innovative Teaching and Learning. *Gagné's nine events of instruction*. In *Instructional guide for university faculty and teaching assistants*. 2020. URL: https://www.niu.edu/citl/resources/guides/instructional-guide.

[16] M. Blake, J. McWaters, and R. Galaski. "The next generation of data-sharing in financial services: Using privacy enhancing techniques to unlock new value". In: *World Economic Forum*. 2019, pp. 1–36. URL: https://www.weforum.org/whitepapers/the-next-generation-of-data-sharing-in-financial-services-using-privacy-enhancing-techniques-to-unlock-new-value/.

[17] The Royal Society. *From privacy to partnership - The role of privacy enhancing technologies in data governance and collaborative analysis*. Online. 2023. URL: https://royalsociety.org/-/media/policy/projects/privacy-enhancing-technologies/From-Privacy-to-Partnership.pdf.

[18] United Nations. *United Nations Guide on Privacy-Enhancing Technologies for Official Statistics*. Online. New York: United Nations Committee of Experts on Big Data and Data Science for Official Statistics, 2023. URL: https://unstats.un.org/bigdata/task-teams/privacy/guide/.

[19] A. Shteyn, K. Kollnig, and C. Inverarity. *Federated learning: an introduction*. Tech. rep. Open Data Institute, 2023. URL: https://www.theodi.org/article/federated-learning-an-introduction-report/.

[20] A. Wood, M. Altman, A. Bembenek, M. Bun, M. Gaboardi, J. Honaker, K. Nissim, D. O'Brien, T. Steinke, and S. Vadhan. "Differential Privacy: A Primer for a Non-Technical Audience". In: *Vanderbilt Journal of Entertainment & Technology Law* 21.17 (2018). Berkman Klein Center Research Publication No. 2019-2. Available at SSRN: https://ssrn.com/abstract=3338027 or http://dx.doi.org/10.2139/ssrn.3338027.

[21] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti. "A Survey on Homomorphic Encryption Schemes: Theory and Implementation". In: 51.4 (2018). ISSN: 0360-0300. DOI: 10.1145/3214303.

[22] Y. Lindell. *Secure Multiparty Computation (MPC)*. Cryptology ePrint Archive, Paper 2020/300. 2020. DOI: 10.1145/3387108. URL: https://eprint.iacr.org/2020/300.

[23] D. Benarroch, L. Brandão, M. Maller, and E. Tromer. *ZKProof Community Reference*. Ed. by zkproof.org. Version 0.3. Updated versions at https://docs.zkproof.org/reference. July 2022. URL: https://docs.zkproof.org/reference.

[24] G. M. Garrido, J. Sedlmeir, Ö. Uludağ, I. S. Alaoui, A. Luckow, and F. Matthes. *Revealing the Landscape of Privacy-Enhancing Technologies in the Context of Data Markets for the IoT: A Systematic Literature Review*. 2022. arXiv: 2107.11905 [cs.CR].

[25] B. A. Kitchenham, D. Budgen, and P. Brereton. *Evidence-Based Software Engineering and Systematic Reviews*. Chapman & Hall/CRC, 2015. ISBN: 1482228653.

[26] F. A. Messmer. "Identification of Educational Needs for the Competent Use of Privacy-Enhancing Technologies". Bachelor's thesis. Munich: Technical University of Munich, 2022.

[27] Food and Agriculture Organization. *E-learning methodologies and good practices: A guide for designing and delivering e-learning solutions from the FAO elearning Academy*. second. Rome, 2021. DOI: 10.4060/i2516e.

[28] C. Dilmegani. *Top 6 Challenges of AI in Healthcare & Overcoming them in 2023*. 2022. URL: https://research.aimultiple.com/challenges-of-ai-in-healthcare/ (visited on 04/26/2023).

[29] S. Jordan, C. Fontaine, and R. Hendricks-Sturrup. "Selecting Privacy-Enhancing Technologies for Managing Health Data Use". In: *Frontiers in Public Health* 10 (2022). DOI: 10.3389/fpubh.2022.814163. URL: https://www.frontiersin.org/articles/10.3389/fpubh.2022.814163.

[30] Coursera. *Machine Learning Models: What They Are and How to Build Them*. 2023. URL: https://www.coursera.org/articles/machine-learning-models (visited on 08/06/2023).

[31] D. C. Nguyen, Q.-V. Pham, P. N. Pathirana, M. Ding, A. Seneviratne, Z. Lin, O. Dobre, and W.-J. Hwang. "Federated learning for smart healthcare: A survey". In: *ACM Computing Surveys (CSUR)* 55.3 (2022), pp. 1–37. DOI: 10.1145/3501296.

[32] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed. "Federated learning: A survey on enabling technologies, protocols, and applications". In: *IEEE Access* 8 (2020), pp. 140699–140725. DOI: 10.1109/ACCESS.2020.3013541.

[33] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. "Federated learning: Challenges, methods, and future directions". In: *IEEE signal processing magazine* 37.3 (2020), pp. 50–60. DOI: 10.1109/MSP.2020.2975749.

[34] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. "Communication-efficient learning of deep networks from decentralized data". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. Vol. 54. PMLR. 2017, pp. 1273–1282. URL: https://proceedings.mlr.press/v54/mcmahan17a.html.

[35] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. "Advances and Open Problems in Federated Learning". In: (2021). arXiv: 1912.04977 [cs.LG].

[36] S. Tao. *A brief introduction to Federated Learning — FL Series Part 1*. 2021. URL: https://openzone.medium.com/a-brief-introduction-to-federated-learning-fl-series-part-1-b81c6ec15fb8 (visited on 04/26/2023).

[37] Talend, Inc. *What are Data Silos?* URL: https://www.talend.com/resources/what-are-data-silos/ (visited on 04/26/2023).

[38] Google Developers. *Descending into ML: Training and Loss*. URL: https://developers.google.com/machine-learning/crash-course/descending-into-ml/training-and-loss (visited on 04/26/2023).

[39] Q. Yang, Y. Liu, T. Chen, and Y. Tong. "Federated machine learning: Concept and applications". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 10.2 (2019), pp. 1–19. DOI: 10.1145/3298981.

[40] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao. "Federated learning in mobile edge networks: A comprehensive survey". In: *IEEE Communications Surveys & Tutorials* 22.3 (2020), pp. 2031–2063. DOI: 10.1109/COMST.2020.2986024.

[41] W. Luping, W. Wei, and L. Bo. "CMFL: Mitigating communication overhead for federated learning". In: *2019 IEEE 39th international conference on distributed computing systems (ICDCS)*. IEEE. 2019, pp. 954–964. DOI: 10.1109/ICDCS.2019.00099.

[42] B. Pfitzner, N. Steckhan, and B. Arnrich. "Federated learning in a medical context: A systematic literature review". In: *ACM Transactions on Internet Technology (TOIT)* 21.2 (2021), pp. 1–31. DOI: 10.1145/3412357.

[43] B. McMahan and D. Ramage. *Federated Learning: Collaborative Machine Learning without Centralized Training Data*. 2017. URL: https://ai.googleblog.com/2017/04/federated-learning-collaborative.html (visited on 08/03/2023).

[44] Gartner. *Gartner Says Digital Ethics is at the Peak of Inflated Expectations in the 2021 Gartner Hype Cycle for Privacy*. 2021. URL: https://www.gartner.com/en/newsroom/press-releases/2021-09-30-gartner-says-digital-ethics-is-at-the-peak-of-inflate (visited on 07/11/2023).

[45] M. Hughes. *Personal info of 31 million people leaked by popular virtual keyboard Ai.type*. 2017. URL: https://thenextweb.com/news/personal-info-31-million-people-leaked-popular-virtual-keyboard-ai-type (visited on 04/26/2023).

[46] K. Hao. *How Apple personalizes Siri without hoovering up your data*. 2019. URL: https://www.technologyreview.com/2019/12/11/131629/apple-ai-personalizes-siri-federated-learning/ (visited on 04/26/2023).

[47] C. Dwork, F. McSherry, K. Nissim, and A. Smith. "Differential Privacy — A Primer for the Perplexed". In: *Joint UNECE/Eurostat work session on statistical data confidentiality* 11 (2011).

[48] C. Dwork and A. Roth. "The algorithmic foundations of differential privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4 (2014), pp. 211–407. DOI: 10.1561/0400000042.

[49] S. Fathima. *What is Differential Privacy?* 2020. URL: https://becominghuman.ai/what-is-differential-privacy-1fd7bf507049 (visited on 05/08/2023).

[50] E. Devaux. *What is Differential Privacy: definition, mechanisms, and examples*. 2022. URL: https://www.statice.ai/post/what-is-differential-privacy-definition-mechanisms-examples (visited on 05/08/2023).

[51] S. Fathima. *Query "Sensitivity" types and effects on Differential Privacy Mechanism*. 2020. URL: https://medium.com/@shaistha24/query-sensitivity-types-and-effects-on-differential-privacy-mechanism-c94fd14b9837 (visited on 05/08/2023).

[52] C. Dwork. "The promise of differential privacy: a tutorial on algorithmic techniques". In: *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science, D (Oct. 2011)*. 2011, pp. 1–2. DOI: 10.1109/FOCS.2011.88.

[53] S. Fathima. *Global vs Local Differential Privacy*. 2020. URL: https://medium.com/@shaistha24/global-vs-local-differential-privacy-56b45eb22168 (visited on 05/08/2023).

[54] S. Fathima. *Differential Privacy Definition*. 2020. URL: https://medium.com/@shaistha24/differential-privacy-definition-bbd638106242 (visited on 05/08/2023).

[55] A. Nguyen. *Understanding Differential Privacy*. 2019. URL: https://towardsdatascience.com/understanding-differential-privacy-85ce191e198a (visited on 05/08/2023).

[56] C. Dwork. "Differential privacy". In: *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II 33*. Springer. 2006, pp. 1–12. DOI: 10.1007/11787006_1.

[57] M. Aitsam. "Differential privacy made easy". In: *2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE)*. IEEE. 2022, pp. 1–7. DOI: 10.48550/arXiv.2201.00099.

[58] C. Dwork. "Differential privacy: A survey of results". In: *Theory and Applications of Models of Computation: 5th International Conference, TAMC 2008, Xi'an, China, April 25-29, 2008. Proceedings 5*. Springer. 2008, pp. 1–19. ISBN: 978-3-540-79228-4. DOI: 10.1007/978-3-540-79228-4_1.

[59] S. Fathima. *Differential Privacy — Noise adding Mechanisms*. 2020. URL: https://medium.com/@shaistha24/differential-privacy-noise-adding-mechanisms-ede242dcbb2e (visited on 05/08/2023).

[60] J. Near, D. Darais, and K. Boeckl. *Differential Privacy for Privacy-Preserving Data Analysis: An Introduction to our Blog Series*. 2020. URL: https://www.nist.gov/blogs/cybersecurity-insights/differential-privacy-privacy-preserving-data-analysis-introduction-our (visited on 05/08/2023).

[61] C. Dilmegani. *Differential Privacy: How It Works, Benefits & Use Cases in 2023*. 2022. URL: https://research.aimultiple.com/differential-privacy/ (visited on 05/08/2023).

[62] Y. Xiao and L. Xiong. "Protecting locations with differential privacy under temporal correlations". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. 2015, pp. 1298–1309. DOI: 10.1145/2810103.2813640.

[63] T. T. Mueller, D. Usynin, J. C. Paetzold, D. Rueckert, and G. Kaissis. *SoK: Differential Privacy on Graph-Structured Data*. 2022. arXiv: 2203.09205 [cs.CR].

[64] M. U. Hassan, M. H. Rehmani, and J. Chen. "Differential Privacy Techniques for Cyber Physical Systems: A Survey". In: *IEEE Communications Surveys & Tutorials* 22.1 (2020), pp. 746–789. DOI: 10.1109/COMST.2019.2944748.

[65] S. L. Garfinkel, J. M. Abowd, and S. Powazek. "Issues encountered deploying differential privacy". In: *Proceedings of the 2018 Workshop on Privacy in the Electronic Society*. 2018, pp. 133–137. DOI: 10.1145/3267323.3268949.

[66] M. B. Hawes. "Implementing Differential Privacy: Seven Lessons From the 2020 United States Census". In: *Harvard Data Science Review* 2.2 (2020). URL: https://hdsr.mitpress.mit.edu/pub/dgg03vo6.

[67] Coursera. *What Is Statistical Analysis? Definition, Types, and Jobs*. 2023. URL: https://www.coursera.org/articles/statistical-analytics (visited on 05/08/2023).

[68] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor. "Federated learning with differential privacy: Algorithms and performance analysis". In: *IEEE Transactions on Information Forensics and Security* 15 (2020), pp. 3454–3469. DOI: 10.1109/TIFS.2020.2988575.

[69] S. Gossett. *What Is Differential Privacy? Can 'Noisy' Data Help Add Anonymity to Public Data Sets?* 2020. URL: https://builtin.com/data-science/differential-privacy (visited on 05/08/2023).

[70] R. Jarmin. *Census Bureau Adopts Cutting Edge Privacy Protections for 2020 Census*. 2019. URL: https://www.census.gov/newsroom/blogs/random-samplings/2019/02/census_bureau_adopts.html (visited on 05/08/2023).

[71] United States Census Bureau. *About the Bureau*. URL: https://www.census.gov/about.html (visited on 05/08/2023).

[72] Apple Inc. *Apple Differential Privacy Technical Overview*. URL: https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf (visited on 05/08/2023).

[73] J. Tang, A. Korolova, X. Bai, X. Wang, and X. Wang. "Privacy Loss in Apple's Implementation of Differential Privacy on MacOS 10.12". In: (2017). arXiv: 1709.02753 [cs.CR].

[74] C. Juvekar, V. Vaikuntanathan, and A. Chandrakasan. *Gazelle: A Low Latency Framework for Secure Neural Network Inference*. 2018. arXiv: 1801.05507 [cs.CR].

[75] N. Dowlin, R. Gilad-Bachrach, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing. *CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy*. Tech. rep. MSR-TR-2016-3. 2016. URL: https://www.microsoft.com/en-us/research/publication/cryptonets-applying-neural-networks-to-encrypted-data-with-high-throughput-and-accuracy/.

[76] IBM. *IBM Security Homomorphic Encryption Services*. Brochure. 2020. URL: https://www.ibm.com/downloads/cas/KQ27PWBO.

[77] R. Lagendijk, Z. Erkin, and M. Barni. "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation". In: *IEEE Signal Processing Magazine* (2013), pp. 82–105. DOI: 10.1109/MSP.2012.2219653.

[78] L. Rosencrance. *Ciphertext*. 2020. URL: https://www.techtarget.com/whatis/definition/ciphertext (visited on 05/23/2023).

[79] E. Maass. *Fully Homomorphic Encryption: Unlocking the Value of Sensitive Data While Preserving Privacy*. 2020. URL: https://securityintelligence.com/posts/fully-homomorphic-encryption-next-step-data-privacy/ (visited on 05/23/2023).

[80] M. D. Ryan. "Cloud computing security: The scientific challenge, and a survey of solutions". In: *Journal of Systems and Software* 86.9 (2013), pp. 2263–2268. ISSN: 0164-1212. DOI: 10.1016/j.jss.2012.12.025.

[81] D. Huynh. *Homomorphic Encryption intro: Part 2: HE landscape and CKKS*. 2020. URL: https://towardsdatascience.com/homomorphic-encryption-intro-part-2-he-landscape-and-ckks-8b32ba5b04dd (visited on 05/23/2023).

[82] D. Huynh. *Homomorphic Encryption intro: Part 1: Overview and use cases*. 2020. URL: https://towardsdatascience.com/homomorphic-encryption-intro-part-1-overview-and-use-cases-a601adcff06c (visited on 05/23/2023).

[83] A. S. Gillis. *Homomorphic Encryption*. 2022. URL: https://www.techtarget.com/searchsecurity/definition/homomorphic-encryption (visited on 05/23/2023).

[84] R. Yackel. *What is homomorphic encryption, and why isn't it mainstream?* 2021. URL: https://www.keyfactor.com/blog/what-is-homomorphic-encryption/ (visited on 05/23/2023).

[85] Optalysys. *Encrypted search using fully homomorphic encryption*. 2021. URL: https://medium.com/optalysys/encrypted-search-using-fully-homomorphic-encryption-4431e987ba40 (visited on 05/23/2023).

[86] A. Wood, K. Najarian, and D. Kahrobaei. "Homomorphic encryption for machine learning in medicine and bioinformatics". In: *ACM Computing Surveys (CSUR)* 53.4 (2020), pp. 1–35. DOI: 10.1145/3394658.

[87] R. Awadallah and A. Samsudin. "Homomorphic encryption for cloud computing and its challenges". In: *2020 IEEE 7th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*. IEEE. 2020, pp. 1–6. DOI: 10.1109/ICETAS51660.2020.9484283.

[88] S. Yakoubov, V. Gadepally, N. Schear, E. Shen, and A. Yerukhimovich. "A survey of cryptographic approaches to securing big-data analytics in the cloud". In: *2014 IEEE High Performance Extreme Computing Conference (HPEC)*. IEEE. 2014, pp. 1–6. DOI: 10.1109/HPEC.2014.7040943.

[89] Chainlink. *Homomorphic Encryption*. 2023. URL: https://blog.chain.link/homomorphic-encryption/ (visited on 05/23/2023).

[90] C. Gentry. "Fully homomorphic encryption using ideal lattices". In: *Proceedings of the forty-first annual ACM symposium on Theory of computing*. 2009, pp. 169–178. DOI: 10.1145/1536414.1536440.

[91] J. Contreras. *Homomorphic encryption with SEAL*. URL: https://learn.microsoft.com/en-us/azure/architecture/solution-ideas/articles/homomorphic-encryption-seal (visited on 05/23/2023).

[92] Homomorphic Encryption Standardization. *Basics of Homomorphic Encryption*. URL: https://homomorphicencryption.org/introduction/ (visited on 05/23/2023).

[93] K. El Makkaoui, A. Ezzati, and A. B. Hssane. "Challenges of using homomorphic encryption to secure cloud computing". In: *2015 International Conference on Cloud Technologies and Applications (CloudTech)*. IEEE. 2015, pp. 1–7. DOI: 10.1109/CloudTech.2015.7337011.

[94] A. Jackson. *20 Use Cases of Homomorphic Encryption Every CISO Must Know*. 2023. URL: https://www.linkedin.com/pulse/20-use-cases-homomorphic-encryption-every-ciso-must-know-jackson-/ (visited on 05/23/2023).

[95] J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos. "Security and Privacy for Cloud-Based IoT: Challenges". In: *IEEE Communications Magazine* 55.1 (2017), pp. 26–33. DOI: 10.1109/MCOM.2017.1600363CM.

[96] A. Lopardo, T. Farrand, A. J. Hall, and A. Benaissa. *What is homomorphic encryption?* 2020. URL: https://blog.openmined.org/what-is-homomorphic-encryption/ (visited on 05/23/2023).

[97] T. H. Nguyen. *5 Impactful Technologies From the Gartner Emerging Technologies and Trends Impact Radar for 2022*. 2021. URL: https://www.gartner.com/en/articles/5-impactful-technologies-from-the-gartner-emerging-technologies-and-trends-impact-radar-for-2022 (visited on 05/23/2023).

[98] C. Cadwalladr and E. Graham-Harrison. *Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach*. 2018. URL: https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election (visited on 05/23/2023).

[99] O. Masters, H. Hunt, E. Steffinlongo, J. Crawford, F. Bergamaschi, M. E. Dela Rosa, C. C. Quini, C. T. Alves, F. de Souza, and D. G. Ferreira. *Towards a Homomorphic Machine Learning Big Data Pipeline for the Financial Services Sector*. Cryptology ePrint Archive, Paper 2019/1113. 2019. URL: https://eprint.iacr.org/2019/1113.

[100] IBM Research. *Top Brazilian Bank Pilots Privacy Encryption Quantum Computers Can't Break*. 2020. URL: https://ibm-research.medium.com/top-brazilian-bank-pilots-privacy-encryption-quantum-computers-cant-break-92ed2695bf14 (visited on 05/23/2023).

[101] K. Holt. *Facebook is reportedly trying to analyze encrypted data without deciphering it*. 2021. URL: https://www.engadget.com/facebook-analyze-encrypted-messages-ad-targeting-175739715.html (visited on 05/23/2023).

[102] P. Martins, L. Sousa, and A. Mariano. "A Survey on Fully Homomorphic Encryption: An Engineering Perspective". In: *ACM Computing Surveys (CSUR)* 50.6 (2017), pp. 1–33. DOI: 10.1145/3124441.

[103] A. Smith. *Facebook is trying to analyse users' encrypted messages without reading their texts, says new report*. 2021. URL: https://www.independent.co.uk/tech/facebook-whatsapp-encryption-instagram-reading-data-b1896579.html (visited on 05/23/2023).

[104] H. Cho, D. J. Wu, and B. Berger. "Secure genome-wide association analysis using multiparty computation". In: *Nature biotechnology* 36.6 (2018), pp. 547–551. DOI: 10.1038/nbt.4108.

[105] L. Csomor. *Multi-Party Computation - Wo Sicherheit Türen öffnet*. 2022. URL: https://www.scip.ch/?labs.20220804 (visited on 06/04/2023).

[106] Panther Team. *A Deep Dive Into Secure Multi-Party Computation (MPC)*. 2022. URL: https://blog.pantherprotocol.io/a-deep-dive-into-secure-multi-party-computation-mpc/ (visited on 06/04/2023).

[107] D. Evans, V. Kolesnikov, and M. Rosulek. *A Pragmatic Introduction to Secure MultiParty Computation*. Version: April 15, 2020. Now Publishers, Inc., 2018. URL: https://www.cs.virginia.edu/~evans/pragmaticmpc/pragmaticmpc.pdf.

[108] C. Zhao, S. Zhao, M. Zhao, Z. Chen, C.-Z. Gao, H. Li, and Y.-a. Tan. "Secure multi-party computation: theory, practice and applications". In: *Information Sciences* 476 (2019), pp. 357–372. DOI: 10.1016/j.ins.2018.10.024.

[109] Inpher. *What is Secure Multiparty Computation*. URL: https://inpher.io/technology/what-is-secure-multiparty-computation/ (visited on 06/04/2023).

[110] C. Dilmegani. *In-Depth Guide Into Secure Multi-Party Computation in 2023*. 2023. URL: https://research.aimultiple.com/secure-multi-party-computation/ (visited on 06/04/2023).

[111] National Institute of Standards and Technology. *Cryptographic primitive*. URL: https://csrc.nist.gov/glossary/term/cryptographic_primitive (visited on 06/04/2023).

[112] Nawazmalla. *What is Secure Multiparty Computation?* URL: https://www.geeksforgeeks.org/what-is-secure-multiparty-computation/ (visited on 06/04/2023).

[113] A. C. Yao. "How to generate and exchange secrets". In: *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*. 1986, pp. 162–167. DOI: 10.1109/SFCS.1986.25.

[114] B. Spector. *What is multi-party computation (MPC)?* 2021. URL: https://www.qredo.com/blog/what-is-multi-party-computation-mpc (visited on 06/04/2023).

[115] N. Kaaniche, M. Laurent, and S. Belguith. "Privacy enhancing technologies for solving the privacy-personalization paradox: Taxonomy and survey". In: *Journal of Network and Computer Applications* 171 (2020), p. 102807. DOI: 10.1016/j.jnca.2020.102807.

[116] Keyless Technologies. *A beginner's guide to Secure Multiparty Computation*. 2020. URL: https://medium.com/@keylesstech/a-beginners-guide-to-secure-multiparty-computation-dc3fb9365458 (visited on 06/04/2023).

[117] Chainlink. *Secure Multi-Party Computation*. 2023. URL: https://blog.chain.link/secure-multi-party-computation-mcp/ (visited on 06/04/2023).

[118] IEEE Digital Privacy. *Applications of Multiparty Computation*. URL: https://digitalprivacy.ieee.org/publications/topics/applications-of-multiparty-computation (visited on 06/04/2023).

[119] Y. Yang, X. Huang, X. Liu, H. Cheng, J. Weng, X. Luo, and V. Chang. "A comprehensive survey on secure outsourced computation and its applications". In: *IEEE Access* 7 (2019), pp. 159426–159465. DOI: 10.1109/ACCESS.2019.2949782.

[120] J. Bringer, H. Chabanne, and A. Patey. "Privacy-preserving biometric identification using secure multiparty computation: An overview and recent trends". In: *IEEE Signal Processing Magazine* 30.2 (2013), pp. 42–52. DOI: 10.1109/MSP.2012.2230218.

[121] M. Hastings, B. Hemenway, D. Noble, and S. Zdancewic. "Sok: General purpose compilers for secure multi-party computation". In: *2019 IEEE symposium on security and privacy (SP)*. IEEE. 2019, pp. 1220–1237. DOI: 10.1109/SP.2019.00028.

[122] N. Volgushev, M. Schwarzkopf, B. Getchell, M. Varia, A. Lapets, and A. Bestavros. "Conclave: secure multi-party computation on big data". In: *Proceedings of the Fourteenth EuroSys Conference 2019*. EuroSys '19. 2019, pp. 1–18. DOI: 10.1145/3302424.3303982.

[123] F. N. Wirth, T. Kussel, A. Müller, K. Hamacher, and F. Prasser. "EasySMPC: a simple but powerful no-code tool for practical secure multiparty computation". In: *BMC bioinformatics* 23.1 (2022), p. 531. DOI: 10.1186/s12859-022-05044-8.

[124] Cambridge Dictionary. *Auction*. URL: https://dictionary.cambridge.org/de/worterbuch/englisch/auction (visited on 06/04/2023).

[125] W. Kenton. *Sealed-Bid Auction Definition, How It Works in Real Estate Sales*. 2022. URL: https://www.investopedia.com/terms/s/sealed-bid-auction.asp (visited on 06/04/2023).

[126] P. Bogetoft, D. L. Christensen, I. Damgård, M. Geisler, T. Jakobsen, M. Krøigaard, J. D. Nielsen, J. B. Nielsen, K. Nielsen, J. Pagter, M. Schwartzbach, and T. Toft. *Multiparty Computation Goes Live*. Cryptology ePrint Archive, Paper 2008/068. 2008. URL: https://eprint.iacr.org/2008/068.

[127] A. C. Yao. "Protocols for secure computations". In: *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*. 1982, pp. 160–164. DOI: 10.1109/SFCS.1982.38.

[128] Gartner. *Understanding Gartner's Hype Cycles*. 2018. URL: https://www.gartner.com/en/documents/3887767 (visited on 06/04/2023).

[129] B. Iannotta. *U.S. Satellite Destroyed in Space Collision*. 2009. URL: https://spacenews.com/u-s-satellite-destroyed-in-space-collision/ (visited on 06/04/2023).

[130] B. Hemenway, S. Lu, R. Ostrovsky, and W. Welser Iv. "High-precision secure computation of satellite collision probabilities". In: *Security and Cryptography for Networks: 10th International Conference, SCN 2016, Amalfi, Italy, August 31–September 2, 2016, Proceedings 10*. Springer. 2016, pp. 169–187. DOI: 10.1007/978-3-319-44618-9_9.

[131] B. Hemenway, W. W. IV, and D. Baiocchi. *Achieving Higher-Fidelity Conjunction Analyses Using Cryptography to Improve Information Sharing*. Santa Monica, CA: RAND Corporation, 2014. URL: https://www.rand.org/pubs/research_reports/RR344.html.

[132] S. Goldwasser, S. Micali, and C. Rackoff. "The Knowledge Complexity of Interactive Proof-Systems". In: *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*. STOC '85. Providence, Rhode Island, USA: Association for Computing Machinery, 1985, pp. 291–304. ISBN: 0897911512. DOI: 10.1145/22145.22178.

[133] Ethereum. *Zero-Knowledge Proofs*. 2023. URL: https://ethereum.org/en/zero-knowledge-proofs/ (visited on 06/18/2023).

[134] S. Liu. "Privacy Protection Revolution: Zero-knowledge Proof". In: *2022 International Conference on Data Analytics, Computing and Artificial Intelligence (ICDACAI)*. IEEE. 2022, pp. 394–397. DOI: 10.1109/ICDACAI57211.2022.00084.

[135] E. Ben-Sasson, A. Chiesa, M. Green, E. Tromer, and M. Virza. "Secure sampling of public parameters for succinct zero knowledge proofs". In: *2015 IEEE Symposium on Security and Privacy*. IEEE. 2015, pp. 287–304. DOI: 10.1109/SP.2015.25.

[136] Chainlink. *Zero-Knowledge Proofs (ZKP)*. 2021. URL: https://blog.chain.link/what-is-a-zero-knowledge-proof-zkp/ (visited on 06/18/2023).

[137] X. Sun, F. R. Yu, P. Zhang, Z. Sun, W. Xie, and X. Peng. "A Survey on Zero-Knowledge Proof in Blockchain". In: *IEEE Network* 35.4 (2021), pp. 198–205. DOI: 10.1109/MNET.011.2000473.

[138] Z. Chen, Y. Jiang, X. Song, and L. Chen. "A Survey on Zero-Knowledge Authentication for Internet of Things". In: *Electronics* 12.5 (2023), p. 1145. DOI: 10.3390/electronics12051145.

[139] Anna. *Guide to zero-knowledge proof systems*. 2022. URL: https://coinloan.io/blog/guide-to-zero-knowledge-proof-systems/ (visited on 06/18/2023).

[140] M. Blum, P. Feldman, and S. Micali. "Non-Interactive Zero-Knowledge and Its Applications". In: STOC '88. Association for Computing Machinery, 1988, pp. 103–112. DOI: 10.1145/62212.62222.

[141] T. Daphne. *Zero-knowledge proofs explained in 3 examples*. 2022. URL: https://www.circularise.com/blogs/zero-knowledge-proofs-explained-in-3-examples (visited on 06/18/2023).

[142] J. Fáwlé. *Zero-Knowledge Proof – How It Works*. 2023. URL: https://hacken.io/discover/zero-knowledge-proof/ (visited on 06/18/2023).

[143] E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev. "Scalable, transparent, and post-quantum secure computational integrity". In: *Cryptology ePrint Archive* (2018). URL: https://eprint.iacr.org/2018/046.

[144] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, and G. Maxwell. "Bulletproofs: Short proofs for confidential transactions and more". In: *2018 IEEE symposium on security and privacy (SP)*. IEEE. 2018, pp. 315–334. DOI: 10.1109/SP.2018.00020.

[145] N. Bitansky, R. Canetti, A. Chiesa, and E. Tromer. "From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again". In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. 2012, pp. 326–349.

[146] Bybit Fintech Limited. *Zero-Knowledge Proof*. 2022. URL: https://learn.bybit.com/glossary/definition-zero-knowledge-proof/ (visited on 06/18/2023).

[147] A. Zunino. *What Are Zero-Knowledge Proofs?* 2023. URL: https://www.forbes.com/sites/forbestechcouncil/2023/02/07/what-are-zero-knowledge-proofs/ (visited on 06/18/2023).

[148] Keyless Technologies. *How Keyless uses zero-knowledge proofs to protect your biometric data*. 2022. URL: https://keyless.io/blog/post/how-keyless-uses-zero-knowledge-proofs-to-protect-your-biometric-data (visited on 06/18/2023).

[149] Metacamp. *Understanding the Basics of Zero Knowledge Proof*. URL: https://www.metacamp.so/insights/understanding-the-basics-of-zero-knowledge-proof (visited on 06/18/2023).

[150] C. Dilmegani. *Zero-Knowledge Proofs: How it Works & Use Cases in 2023*. 2022. URL: https://research.aimultiple.com/zero-knowledge-proofs/ (visited on 06/18/2023).

[151] D. Kotecha. *Zero Knowledge Proof*. 2021. URL: https://dhruvilkotecha.medium.com/zero-knowledge-proof-23bd86b70551 (visited on 06/18/2023).

[152] E. Seker. *Zero-Knowledge Proofs (ZKPs)*. 2021. URL: https://medium.com/codex/zero-knowledge-proofs-zkps-26d26dc2eb26 (visited on 06/18/2023).

[153] M. F. Esgin, R. K. Zhao, R. Steinfeld, J. K. Liu, and D. Liu. "MatRiCT: efficient, scalable and post-quantum blockchain confidential transactions protocol". In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2019, pp. 567–584. DOI: 10.1145/3319535.3354200.

[154] International Organization for Standardization. *ISO/IEC WD 27565.3 – Guidelines on privacy preservation based on zero knowledge proofs*. 2023. URL: https://www.iso.org/standard/80398.html (visited on 06/18/2023).

[155] Z. Zhang, K. Yang, X. Hu, and Y. Wang. "Practical anonymous password authentication and TLS with anonymous client authentication". In: *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 2016, pp. 1179–1191. DOI: 10.1145/2976749.2978354.

[156] D. Sanjith. *Zero Knowledge Proof (ZKP): A brief Overview*. 2023. URL: https://www.techlab.cdc.gov/index.php/2023/04/01/zero-knowledge-proof-zkp/ (visited on 06/18/2023).

[157] M. Blum, A. De Santis, S. Micali, and G. Persiano. "Noninteractive Zero-Knowledge". In: *SIAM Journal on Computing* 20.6 (1991), pp. 1084–1118. DOI: 10.1137/0220068.

[158] F. Gaehtgens. *Hype Cycle for Digital Identity, 2022*. 2022. URL: https://www.gartner.com/doc/reprints?id=1-2AUBQDQC&ct=220815&st=sb (visited on 06/18/2023).

[159] S. Gunaratna. *LinkedIn: 2012 data breach much worse than we thought*. 2016. URL: https://www.cbsnews.com/news/linkedin-2012-data-breach-hack-much-worse-than-we-thought-passwords-emails/ (visited on 06/18/2023).

[160] F. O'Sullivan. *Is Crypto Anonymous in 2023 & Is It More Traceable Than Cash?* 2023. URL: https://www.cloudwards.net/is-crypto-anonymous/ (visited on 06/18/2023).